Cisco ONS 15310-CL and Cisco ONS 15310-MA Ethernet Card Software Feature and Configuration Guide

Cisco IOS Release 12.2(28)SV
CTC and Documentation Release 8.5
June 2009

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Text Part Number: 78-18133-01

Cisco ONS 15310-CL and Cisco ONS 15310-MA Ethernet Card Software Feature and Configuration Guide, Release 8.5

CONTENTS

FIGURES

TABLES

Preface

Note The terms "Unidirectional Path Switched Ring" and "UPSR" may appear in Cisco literature. These terms do not refer to using Cisco ONS 15xxx products in a unidirectional path switched ring configuration. Rather, these terms, as well as "Path Protected Mesh Network" and "PPMN," refer generally to Cisco's path protection feature, which may be used in any topological network configuration. Cisco does not recommend using its path protection feature in any particular topological network configuration.

This section provides the following information:

Revision History

Date	Notes
July 2008	Modified a statement in the “Flow Control Pause and QoS” section of Chapter 12, Configuring Quality of Service.
September 2008	Updated the section “CE-100T-8 VCAT Characteristics” in Chapter 17, CE-100T-8 Ethernet Operation.
December 2008	Added a new section “Load Balancing on the ML-Series Cards” in Chapter 9, Configuring Link Aggregation on the ML-Series Cards”.

Date	Notes
January 2009	Added the following sections in Chapter 11, Configuring Quality of Service on the ML-Series Card: • QoS not Configured on Egress • ML-Series Egress Bandwidth Example • Added a new bullet point in the "IP SLA Restrcitions on the ML-Series" section. • Added Tables 9-1 and 9-2 and updated Table 4 in the "Load Balancing on ML-Series Cards" section of Chapter 9, Configuring Link Aggregation on the ML-Series Cards.
February 2009	Added a note in the "Ring Wrapping" section of Chapter 15, Configuring Resilient Packet Ring on the ML-Series Card.
June 2009	Updated the sections "RMON" and "SNMP" in Chapter 1, Overview of the ML-Series Cards.

Document Objectives

This guide covers the software features and operations of the ML-100T-8 and the CE-100T-8 Ethernet cards for the Cisco ONS 15310-CL and the Cisco ONS 15310-MA. It explains software features and configuration for Cisco IOS on the ML-Series card. It also explains software feature and configuration for Cisco Transport Controller (CTC) on the CE-100T-8 card. The CE-100T-8 card is also available as a card for the Cisco ONS 15454 and Cisco ONS 15454 SDH. This version of the card is described in the *Cisco ONS 15454 and Cisco ONS 15454 SDH Ethernet Card Software Feature and Configuration Guide*. Use this guide in conjunction with the appropriate publications listed in the Related Documentation section.

Audience

To use the ML-Series card chapters of this publication, you should be familiar with Cisco IOS and preferably have technical networking background and experience. To use the CE-100T-8 card chapter of this publication, you should be familiar with CTC and preferably have technical networking background and experience.

Related Documentation

Use the *Cisco ONS 15310-CL and Cisco ONS 15310-MA Ethernet Card Software Feature and Configuration Guide R8.5* in conjunction with the following general ONS 15310-CL and ONS 15310-MA system publications:

- To install, turn up, provision, and maintain a Cisco ONS 15310-CL or Cisco ONS 15310-MA node and network, refer to the *Cisco ONS 15310-CL and Cisco ONS 15310-MA Procedure Guide*.
- For alarm clearing, general troubleshooting procedures, transient conditions, and error messages for a Cisco ONS 15310-CL and Cisco ONS 15310-MA card, node, or network, refer to the *Cisco ONS 15310-CL and Cisco ONS 15310-MA Troubleshooting Guide*.

- For detailed reference information about Cisco ONS 15310-CL or Cisco ONS 15310-MA cards, nodes, and networks, refer to the *Cisco ONS 15310-CL and Cisco ONS 15310-MA Reference Manual*.

The ML-Series card employs the Cisco IOS Modular QoS CLI (MQC). For more information on general MQC configuration, refer to the following Cisco IOS documents:

- *Cisco IOS Quality of Service Solutions Configuration Guide, Release 12.2*
- *Cisco IOS Quality of Service Solutions Command Reference, Release 12.2*
- The ML-Series card employs Cisco IOS 12.2. For more general information on Cisco IOS 12.2, refer to the extensive Cisco IOS documentation at http://www.cisco.com.

For an update on End-of-Life and End-of-Sale notices, refer to http://cisco.com/en/US/products/hw/optical/ps2001/prod_eol_notices_list.html.

Document Conventions

This publication uses the following conventions:

Convention	Application
boldface	Commands and keywords in body text.
italic	Command input that is supplied by the user.
[ ]	Keywords or arguments that appear within square brackets are optional.
{ x | x | x }	A choice of keywords (represented by x) appears in braces separated by vertical bars. The user must select one.
Ctrl	The control key. For example, where Ctrl + D is written, hold down the Control key while pressing the D key.
`screen font`	Examples of information displayed on the screen.
`boldface screen font`	Examples of information that the user must enter.
< >	Command parameters that must be replaced by module-specific codes.

Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the document.

Caution Means *reader be careful*. In this situation, the user might do something that could result in equipment damage or loss of data.

Warning **IMPORTANT SAFETY INSTRUCTIONS**

This warning symbol means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Use the statement number provided at the end of each warning to locate its translation in the translated safety warnings that accompanied this device. Statement 1071

SAVE THESE INSTRUCTIONS

Waarschuwing **BELANGRIJKE VEILIGHEIDSINSTRUCTIES**

Dit waarschuwingssymbool betekent gevaar. U verkeert in een situatie die lichamelijk letsel kan veroorzaken. Voordat u aan enige apparatuur gaat werken, dient u zich bewust te zijn van de bij elektrische schakelingen betrokken risico's en dient u op de hoogte te zijn van de standaard praktijken om ongelukken te voorkomen. Gebruik het nummer van de verklaring onderaan de waarschuwing als u een vertaling van de waarschuwing die bij het apparaat wordt geleverd, wilt raadplegen.

BEWAAR DEZE INSTRUCTIES

Varoitus **TÄRKEITÄ TURVALLISUUSOHJEITA**

Tämä varoitusmerkki merkitsee vaaraa. Tilanne voi aiheuttaa ruumiillisia vammoja. Ennen kuin käsittelet laitteistoa, huomioi sähköpiirien käsittelemiseen liittyvät riskit ja tutustu onnettomuuksien yleisiin ehkäisytapoihin. Turvallisuusvaroitusten käännökset löytyvät laitteen mukana toimitettujen käännettyjen turvallisuusvaroitusten joukosta varoitusten lopussa näkyvien lausuntonumeroiden avulla.

SÄILYTÄ NÄMÄ OHJEET

Attention **IMPORTANTES INFORMATIONS DE SÉCURITÉ**

Ce symbole d'avertissement indique un danger. Vous vous trouvez dans une situation pouvant entraîner des blessures ou des dommages corporels. Avant de travailler sur un équipement, soyez conscient des dangers liés aux circuits électriques et familiarisez-vous avec les procédures couramment utilisées pour éviter les accidents. Pour prendre connaissance des traductions des avertissements figurant dans les consignes de sécurité traduites qui accompagnent cet appareil, référez-vous au numéro de l'instruction situé à la fin de chaque avertissement.

CONSERVEZ CES INFORMATIONS

Warnung **WICHTIGE SICHERHEITSHINWEISE**

Dieses Warnsymbol bedeutet Gefahr. Sie befinden sich in einer Situation, die zu Verletzungen führen kann. Machen Sie sich vor der Arbeit mit Geräten mit den Gefahren elektrischer Schaltungen und den üblichen Verfahren zur Vorbeugung vor Unfällen vertraut. Suchen Sie mit der am Ende jeder Warnung angegebenen Anweisungsnummer nach der jeweiligen Übersetzung in den übersetzten Sicherheitshinweisen, die zusammen mit diesem Gerät ausgeliefert wurden.

BEWAHREN SIE DIESE HINWEISE GUT AUF.

Avvertenza **IMPORTANTI ISTRUZIONI SULLA SICUREZZA**

Questo simbolo di avvertenza indica un pericolo. La situazione potrebbe causare infortuni alle persone. Prima di intervenire su qualsiasi apparecchiatura, occorre essere al corrente dei pericoli relativi ai circuiti elettrici e conoscere le procedure standard per la prevenzione di incidenti. Utilizzare il numero di istruzione presente alla fine di ciascuna avvertenza per individuare le traduzioni delle avvertenze riportate in questo documento.

CONSERVARE QUESTE ISTRUZIONI

Advarsel **VIKTIGE SIKKERHETSINSTRUKSJONER**

Dette advarselssymbolet betyr fare. Du er i en situasjon som kan føre til skade på person. Før du begynner å arbeide med noe av utstyret, må du være oppmerksom på farene forbundet med elektriske kretser, og kjenne til standardprosedyrer for å forhindre ulykker. Bruk nummeret i slutten av hver advarsel for å finne oversettelsen i de oversatte sikkerhetsadvarslene som fulgte med denne enheten.

TA VARE PÅ DISSE INSTRUKSJONENE

Aviso **INSTRUÇÕES IMPORTANTES DE SEGURANÇA**

Este símbolo de aviso significa perigo. Você está em uma situação que poderá ser causadora de lesões corporais. Antes de iniciar a utilização de qualquer equipamento, tenha conhecimento dos perigos envolvidos no manuseio de circuitos elétricos e familiarize-se com as práticas habituais de prevenção de acidentes. Utilize o número da instrução fornecido ao final de cada aviso para localizar sua tradução nos avisos de segurança traduzidos que acompanham este dispositivo.

GUARDE ESTAS INSTRUÇÕES

¡Advertencia! **INSTRUCCIONES IMPORTANTES DE SEGURIDAD**

Este símbolo de aviso indica peligro. Existe riesgo para su integridad física. Antes de manipular cualquier equipo, considere los riesgos de la corriente eléctrica y familiarícese con los procedimientos estándar de prevención de accidentes. Al final de cada advertencia encontrará el número que le ayudará a encontrar el texto traducido en el apartado de traducciones que acompaña a este dispositivo.

GUARDE ESTAS INSTRUCCIONES

Varning! **VIKTIGA SÄKERHETSANVISNINGAR**

Denna varningssignal signalerar fara. Du befinner dig i en situation som kan leda till personskada. Innan du utför arbete på någon utrustning måste du vara medveten om farorna med elkretsar och känna till vanliga förfaranden för att förebygga olyckor. Använd det nummer som finns i slutet av varje varning för att hitta dess översättning i de översatta säkerhetsvarningar som medföljer denna anordning.

SPARA DESSA ANVISNINGAR

FONTOS BIZTONSÁGI ELOÍRÁSOK

Ez a figyelmezeto jel veszélyre utal. Sérülésveszélyt rejto helyzetben van. Mielott bármely berendezésen munkát végezte, legyen figyelemmel az elektromos áramkörök okozta kockázatokra, és ismerkedjen meg a szokásos balesetvédelmi eljárásokkal. A kiadványban szereplo figyelmeztetések fordítása a készülékhez mellékelt biztonsági figyelmeztetések között található; a fordítás az egyes figyelmeztetések végén látható szám alapján keresheto meg.

ORIZZE MEG EZEKET AZ UTASÍTÁSOKAT!

Предупреждение **ВАЖНЫЕ ИНСТРУКЦИИ ПО СОБЛЮДЕНИЮ ТЕХНИКИ БЕЗОПАСНОСТИ**

Этот символ предупреждения обозначает опасность. То есть имеет место ситуация, в которой следует опасаться телесных повреждений. Перед эксплуатацией оборудования выясните, каким опасностям может подвергаться пользователь при использовании электрических цепей, и ознакомьтесь с правилами техники безопасности для предотвращения возможных несчастных случаев. Воспользуйтесь номером заявления, приведенным в конце каждого предупреждения, чтобы найти его переведенный вариант в переводе предупреждений по безопасности, прилагаемом к данному устройству.

СОХРАНИТЕ ЭТИ ИНСТРУКЦИИ

警告 重要的安全性说明

此警告符号代表危险。您正处于可能受到严重伤害的工作环境中。在您使用设备开始工作之前，必须充分意识到触电的危险，并熟练掌握防止事故发生的标准工作程序。请根据每项警告结尾提供的声明号码来找到此设备的安全性警告说明的翻译文本。

请保存这些安全性说明

警告 安全上の重要な注意事項

「危険」の意味です。人身事故を予防するための注意事項が記述されています。装置の取り扱い作業を行うときは、電気回路の危険性に注意し、一般的な事故防止策に留意してください。警告の各国語版は、各注意事項の番号を基に、装置に付属の「Translated Safety Warnings」を参照してください。

これらの注意事項を保管しておいてください。

주의 중요 안전 지침

이 경고 기호는 위험을 나타냅니다. 작업자가 신체 부상을 일으킬 수 있는 위험한 환경에 있습니다. 장비에 작업을 수행하기 전에 전기 회로와 관련된 위험을 숙지하고 표준 작업 관례를 숙지하여 사고를 방지하십시오. 각 경고의 마지막 부분에 있는 경고문 번호를 참조하여 이 장치와 함께 제공되는 번역된 안전 경고문에서 해당 번역문을 찾으십시오.

이 지시 사항을 보관하십시오.

Aviso **INSTRUÇÕES IMPORTANTES DE SEGURANÇA**

Este símbolo de aviso significa perigo. Você se encontra em uma situação em que há risco de lesões corporais. Antes de trabalhar com qualquer equipamento, esteja ciente dos riscos que envolvem os circuitos elétricos e familiarize-se com as práticas padrão de prevenção de acidentes. Use o número da declaração fornecido ao final de cada aviso para localizar sua tradução nos avisos de segurança traduzidos que acompanham o dispositivo.

GUARDE ESTAS INSTRUÇÕES

Advarsel **VIGTIGE SIKKERHEDSANVISNINGER**

Dette advarselssymbol betyder fare. Du befinder dig i en situation med risiko for legemesbeskadigelse. Før du begynder arbejde på udstyr, skal du være opmærksom på de involverede risici, der er ved elektriske kredsløb, og du skal sætte dig ind i standardprocedurer til undgåelse af ulykker. Brug erklæringsnummeret efter hver advarsel for at finde oversættelsen i de oversatte advarsler, der fulgte med denne enhed.

GEM DISSE ANVISNINGER

تحذير **إرشادات الأمان الهامة**

يوضح رمز التحذير هذا وجود خطر. وهذا يعني أنك متواجد في مكان قد ينتج عنه التعرض لإصابات. قبل بدء العمل، احذر مخاطر التعرض للصدمات الكهربائية وكن على علم بالإجراءات القياسية للحيلولة دون وقوع أي حوادث. استخدم رقم البيان الموجود في أخر كل تحذير لتحديد مكان ترجمته داخل تحذيرات الأمان المترجمة التي تأتي مع الجهاز.
قم بحفظ هذه الإرشادات

Upozorenje **VAŽNE SIGURNOSNE NAPOMENE**

Ovaj simbol upozorenja predstavlja opasnost. Nalazite se u situaciji koja može prouzročiti tjelesne ozljede. Prije rada s bilo kojim uređajem, morate razumjeti opasnosti vezane uz električne sklopove, te biti upoznati sa standardnim načinima izbjegavanja nesreća. U prevedenim sigurnosnim upozorenjima, priloženima uz uređaj, možete prema broju koji se nalazi uz pojedino upozorenje pronaći i njegov prijevod.

SAČUVAJTE OVE UPUTE

Upozornění **DŮLEŽITÉ BEZPEČNOSTNÍ POKYNY**

Tento upozorňující symbol označuje nebezpečí. Jste v situaci, která by mohla způsobit nebezpečí úrazu. Před prací na jakémkoliv vybavení si uvědomte nebezpečí související s elektrickými obvody a seznamte se se standardními opatřeními pro předcházení úrazům. Podle čísla na konci každého upozornění vyhledejte jeho překlad v přeložených bezpečnostních upozorněních, která jsou přiložena k zařízení.

USCHOVEJTE TYTO POKYNY

Προειδοποίηση

ΣΗΜΑΝΤΙΚΕΣ ΟΔΗΓΙΕΣ ΑΣΦΑΛΕΙΑΣ

Αυτό το προειδοποιητικό σύμβολο σημαίνει κίνδυνο. Βρίσκεστε σε κατάσταση που μπορεί να προκαλέσει τραυματισμό. Πριν εργαστείτε σε οποιοδήποτε εξοπλισμό, να έχετε υπόψη σας τους κινδύνους που σχετίζονται με τα ηλεκτρικά κυκλώματα και να έχετε εξοικειωθεί με τις συνήθεις πρακτικές για την αποφυγή ατυχημάτων. Χρησιμοποιήστε τον αριθμό δήλωσης που παρέχεται στο τέλος κάθε προειδοποίησης, για να εντοπίσετε τη μετάφρασή της στις μεταφρασμένες προειδοποιήσεις ασφαλείας που συνοδεύουν τη συσκευή.

ΦΥΛΑΞΤΕ ΑΥΤΕΣ ΤΙΣ ΟΔΗΓΙΕΣ

אזהרה

הוראות בטיחות חשובות

סימן אזהרה זה מסמל סכנה. אתה נמצא במצב העלול לגרום לפציעה. לפני שתעבוד עם ציוד כלשהו, עליך להיות מודע לסכנות הכרוכות במעגלים חשמליים ולהכיר את הנהלים המקובלים למניעת תאונות. השתמש במספר ההוראה המסופק בסופה של כל אזהרה כד לאתר את התרגום באזהרות הבטיחות המתורגמות שמצורפות להתקן.

שמור הוראות אלה

Opomena

ВАЖНИ БЕЗБЕДНОСНИ НАПАТСТВИЈА

Симболот за предупредување значи опасност. Се наоѓате во ситуација што може да предизвика телесни повреди. Пред да работите со опремата, бидете свесни за ризикот што постои кај електричните кола и треба да ги познавате стандардните постапки за спречување на несреќни случаи. Искористете го бројот на изјавата што се наоѓа на крајот на секое предупредување за да го најдете неговиот период во преведените безбедносни предупредувања што се испорачани со уредот.

ЧУВАЈТЕ ГИ ОВИЕ НАПАТСТВИЈА

Ostrzeżenie

WAŻNE INSTRUKCJE DOTYCZĄCE BEZPIECZEŃSTWA

Ten symbol ostrzeżenia oznacza niebezpieczeństwo. Zachodzi sytuacja, która może powodować obrażenia ciała. Przed przystąpieniem do prac przy urządzeniach należy zapoznać się z zagrożeniami związanymi z układami elektrycznymi oraz ze standardowymi środkami zapobiegania wypadkom. Na końcu każdego ostrzeżenia podano numer, na podstawie którego można odszukać tłumaczenie tego ostrzeżenia w dołączonym do urządzenia dokumencie z tłumaczeniami ostrzeżeń.

NINIEJSZE INSTRUKCJE NALEŻY ZACHOWAĆ

Upozornenie

DÔLEŽITÉ BEZPEČNOSTNÉ POKYNY

Tento varovný symbol označuje nebezpečenstvo. Nachádzate sa v situácii s nebezpečenstvom úrazu. Pred prácou na akomkoľvek vybavení si uvedomte nebezpečenstvo súvisiace s elektrickými obvodmi a oboznámte sa so štandardnými opatreniami na predchádzanie úrazom. Podľa čísla na konci každého upozornenia vyhľadajte jeho preklad v preložených bezpečnostných upozorneniach, ktoré sú priložené k zariadeniu.

USCHOVAJTE SI TENTO NÁVOD

Obtaining Optical Networking Information

This section contains information that is specific to optical networking products. For information that pertains to all of Cisco, refer to the Obtaining Documentation, Obtaining Support, and Security Guidelines section.

Where to Find Safety and Warning Information

For safety and warning information, refer to the *Cisco Optical Transport Products Safety and Compliance Information* document that accompanied the product. This publication describes the international agency compliance and safety information for the Cisco ONS 15454 system. It also includes translations of the safety warnings that appear in the ONS 15454 system documentation.

Cisco Optical Networking Product Documentation CD-ROM

Optical networking-related documentation, including Cisco ONS 15xxx product documentation, is available in a CD-ROM package that ships with your product. The Optical Networking Product Documentation CD-ROM is updated periodically and may be more current than printed documentation.

Obtaining Documentation, Obtaining Support, and Security Guidelines

For information on obtaining documentation, obtaining support, providing documentation feedback, security guidelines, and also recommended aliases and general Cisco documents, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

CHAPTER 1

Overview of the ML-Series Card

This chapter provides an overview of the ML-100T-8 card for Cisco ONS 15310-CL and the Cisco ONS 15310-MA. It lists Ethernet and SONET capabilities and Cisco IOS and Cisco Transport Controller (CTC) software features, with brief descriptions of selected features.

The CE-100T-8 card for the Cisco ONS 15310-CL and the Cisco ONS 15310-MA is covered in Chapter 17, "CE-100T-8 Ethernet Operation." For Ethernet card specifications, refer to the *Cisco ONS 15454 Reference Manual*. For step-by-step Ethernet card circuit configuration, hard-reset, and soft-reset procedures, refer to the *Cisco ONS 15454 Procedure Guide*. Refer to the *Cisco ONS SONET TL1 Command Guide* for TL1 provisioning commands. For specific details on ONS 15310-CL Ethernet card interoperability with other ONS platforms, refer to the "POS on ONS Ethernet Cards" chapter of the *Cisco ONS 15454 and Cisco ONS 15454 SDH Ethernet Card Software Feature and Configuration Guide*.

This chapter contains the following major sections:

ML-Series Card Description

The ML-Series card is a module in the Cisco ONS 15310-CL and the Cisco ONS 15310-MA. It is an independent Fast Ethernet switch with eight RJ-45 interfaces. The ML-Series card uses Cisco IOS Release 12.2(28)SV, and the Cisco IOS command-line interface (CLI) is the primary user interface for the ML-Series card. Most configuration for the card, such as Ethernet and packet-over-SONET (POS) port provisioning, bridging, VLAN, and Quality of Service (QoS), can be done only with the Cisco IOS CLI.

However, CTC—the ONS 15310-CL graphical user interface (GUI)—and Transaction Language One (TL1) also support the ML-Series card. SONET circuits must be configured through CTC or TL1 and cannot be provisioned through Cisco IOS. CTC also offers ML-Series card status information, SONET alarm management, Cisco IOS Telnet session initialization, provisioning, inventory, and other standard functions.

The ML-Series card features two virtual ports, which function in a manner similar to OC-N card ports. The SONET circuits are provisioned through CTC in the same manner as standard OC-N circuits.

For detailed card specifications, refer to the *Cisco ONS 15454 Reference Manual*.

ML-Series Feature List

The ML-100T-8 has the following features:

- Layer 1 data features:
  - 10/100BASE-TX half-duplex and full-duplex data transmission
  - IEEE 802.3x compliant flow control
- SONET features:
  - High-level data link control (HDLC) or frame-mapped generic framing procedure (GFP-F) framing mechanisms for POS
  - GFP-F supports LEX (default), Cisco HDLC, and Point-to-Point Protocol/Bridging Control Protocol (PPP/BCP) encapsulation for POS
  - HDLC framing supports LEX encapsulation only
  - Two POS virtual ports
  - Virtual concatenated (VCAT) circuits with Link Capacity Adjustment Scheme (LCAS) or without LCAS
  - ONS 15310 ML-Series LCAS is compatible with ONS 15454 ML-Series SW-LCAS
- Layer 2 bridging features:
  - Transparent bridging
  - MAC address learning, aging, and switching by hardware
  - Protocol tunneling
  - Multiple Spanning Tree (MST) protocol tunneling
  - 255 active bridge group maximum
  - 8,000 MAC address maximum per card
  - Integrated routing and bridging (IRB)
  - IEEE 802.1P/Q-based VLAN trunking
  - IEEE 802.1Q VLAN tunneling
  - IEEE 802.1D Spanning Tree Protocol (STP) and IEEE 802.1W Rapid Spanning Tree Protocol (RSTP)
  - IEEE 802.1D STP instance per bridge group
  - Resilient packet ring (RPR)
  - VLAN-transparent and VLAN-specific services (Ethernet Relay Multipoint Service [ERMS])
- Fast EtherChannel (FEC) features:
  - Bundling of up to four Fast Ethernet ports
  - Load sharing based on source and destination IP addresses of unicast packets
  - Load sharing for bridge traffic based on MAC addresses
  - IRB
  - IEEE 802.1Q trunking
  - Up to 4 active FEC port channels
- POS channel:

  - Bundling the two POS ports
  - LEX encapsulation only
  - IRB
  - IEEE 802.1Q trunking
- Layer 3 static routing:
  - Default routes
  - IP unicast and multicast forwarding
  - Reverse Path Forwarding (RPF) multicast (not RPF unicast)
  - Load balancing among equal cost paths based on source and destination IP addresses
  - Up to 350 IP routes per card
  - Up to 350 IP hosts per card
  - IRB routing mode support
- QoS features:
  - Multicast priority queuing classes
  - Service level agreements (SLAs) with 1-Mbps granularity
  - Input policing
  - Guaranteed bandwidth (weighted round-robin [WDRR] plus strict priority scheduling)
  - Low latency queuing support for unicast voice over IP (VoIP)
  - Class of service (CoS) based on Layer 2 priority, VLAN ID, Layer 3 Type of Service/DiffServ Code Point (TOS/DSCP), and port
  - CoS-based packet statistics
  - Up to 350 QoS entries per card
  - Up to 350 policers per card
  - IP SLA network monitoring using Cisco IP SLA (formerly Cisco Service Assurance Agent)
- Security features
  - Cisco IOS login enhancements
  - Secure Shell connection (SSH Version 2)
  - Disabled console port
  - Authentication, Authorization, and Accounting/Remote Authentication Dial-In User Service (AAA/RADIUS) stand alone mode
  - AAA/RADIUS relay mode
- Additional protocols:
  - Cisco Discovery Protocol (CDP) support on Ethernet ports
  - Dynamic Host Configuration Protocol (DHCP) relay
  - Hot Standby Router Protocol (HSRP) over 10/100 Ethernet, FEC and Bridge Group Virtual Interface (BVI)
  - Internet Control Message Protocol (ICMP)
- Management features:

  - Cisco IOS Release 12.2(28)SV
  - CTC
  - Remote monitoring (RMON)
  - Simple Network Management Protocol (SNMP)
  - TL1
- System features:
  - Network Equipment Building Systems 3 (NEBS3) compliant
- CTC features:
  - Standard synchronous transport signal (STS) and VCAT circuit provisioning for POS virtual ports
  - SONET alarm reporting for path alarms and other ML-Series card specific alarms
  - Raw port statistics
  - Standard inventory and card management functions
  - J1 path trace
  - Cisco IOS CLI Telnet sessions from CTC
  - Cisco IOS startup configuration file management from CTC

Key ML-Series Features

This section describes selected key features and their implementation on the ML-Series cards.

Cisco IOS

Cisco IOS controls the data functions of the ML-Series cards. Users cannot update the ML-Series Cisco IOS image in the same manner as the Cisco IOS system image on a Cisco Catalyst Series. An ML-Series Cisco IOS image upgrade is available only as part of the Cisco ONS 15310-CL or the Cisco ONS 15310-MA software release and accomplished only through CTC or TL1. The image is not available for download or shipped separately.

GFP-F Framing

GFP defines a standard-based mapping of different types of services onto SONET/SDH. The ML-Series and CE-Series support frame-mapped GFP (GFP-F), which is the protocol data unit (PDU)-oriented client signal adaptation mode for GFP. GFP-F maps one variable length data packet onto one GFP packet.

GFP is composed of common functions and payload specific functions. Common functions are those shared by all payloads. Payload-specific functions are different depending on the payload type. GFP is detailed in the ITU recommendation G.7041.

Link Aggregation (FEC and POS)

The ML-Series offers Fast EtherChannel and POS channel link aggregation. Link aggregation groups multiple ports into a larger logical port and provides resiliency during the failure of any individual ports. The ML-Series supports a maximum of four Ethernet ports in Fast EtherChannel, and two SONET virtual ports in POS channel. POS channel is only supported with LEX encapsulation.

Traffic flows map to individual ports based on MAC source address (SA)/destination address (DA) for bridged packets and IP SA/DA for routed packets. There is no support for policing or class-based packet priorities when link aggregation is configured.

RMON

The ML-Series card features RMON that allows network operators to monitor the health of the network with an NMS. ONG RMON is recommended for the ML-100T-8. The ONG RMON contains the statistics, history, alarms, and events MIB groups from the standard RMON MIB. The standard Cisco IOS RMON is also available. A user can access RMON threshold provisioning through TL1 or CTC. For more information on RMON, refer to the "SNMP Remote Monitoring" section in "SNMP" chapter of the *Cisco ONS 15310-CL and Cisco ONS 15310-MA Reference Manual*.

RPR

RPR is an emerging network architecture designed for metro fiber ring networks. This new MAC protocol is designed to overcome the limitations of STP, RSTP, and SONET in packet-based networks. RPR convergence times are comparable to SONET and much faster than STP or RSTP. RPR operates at the Layer 2 level and is compatible with Ethernet and protected or unprotected SONET circuits.

SNMP

The Cisco ONS 15310-CL, the Cisco ONS 15310-MA, and the ML-Series cards have SNMP agents and support SNMP Version 1 (SNMPv1) and SNMP Version 2c (SNMPv2c) sets and traps. The Cisco ONS 15310-CL and the Cisco ONS 15310-MA accept, validate, and forward get/getNext/set requests to the ML-Series through a proxy agent. Responses from the ML-Series are relayed by the Cisco ONS 15310-CL and the Cisco ONS 15310-MA to the requesting SNMP agents.

The ML-Series card SNMP support includes:

- STP traps from Bridge-MIB (RFC 1493)
- Authentication traps from RFC 1157
- Export of QoS statistics through the CISCO-PORT-QOS-MIB extension

For more information on how the ONS 15310-CL implements SNMP, refer to the "SNMP" chapter of the *Cisco ONS 15310-CL and Cisco ONS 15310-MA Reference Manual*. For more information on specific MIBs, refer to the Cisco SNMP Object Navigator at http://www.cisco.com.

TL1

TL1 on the ML-Series cards can be used for card inventory, fault and alarm management, card provisioning, and retrieval of status information for both data and SONET ports. TL1 can also be used to provision SONET STS circuits and transfer a Cisco IOS startup configuration file to the card memory. For specific TL1 commands and general TL1 information, refer to the *Cisco ONS SONET TL1 Command Guide*.

CHAPTER 2

CTC Operations on the ML-Series Card

This chapter covers Cisco Transport Controller (CTC) operation of the ML-Series card. All operations described in the chapter take place at the card-level view of CTC. CTC shows provisioning information and statistics for both the Ethernet and packet-over-SONET (POS) ports of the ML-Series card. For the ML-Series cards, CTC manages SONET alarms and provisions STS circuits in the same manner as other Cisco ONS 15310-CL and Cisco ONS 15310-MA SONET traffic.

Use CTC to load a Cisco IOS configuration file or to open a Cisco IOS command-line interface (CLI) session. See Chapter 3, "Initial Configuration of the ML-Series Card."

This chapter contains the following major sections:

Displaying ML-Series POS Statistics in CTC

The POS statistics window lists POS port-level statistics. Display the CTC card view for the ML-Series card and click the **Performance > POS Ports** tabs to display the window.

Table 2-1 describes the buttons in the POS Ports window.

Table 2-1 ML-Series POS Statistics Fields and Buttons

Button	Description
Refresh	Manually refreshes the statistics.

Table 2-1 ML-Series POS Statistics Fields and Buttons

Button	Description
Baseline	Resets the software counters (in that particular CTC client only) temporarily to zero without affecting the actual statistics on the card. From that point on, only counters displaying the change from the temporary baseline are displayed by this CTC client. These new baselined counters are shown only as long as the user displays the Performance window. If the user navigates to another CTC window and comes back to the Performance window, the true actual statistics retained by the card are shown.
Auto-Refresh	Sets a time interval for the automatic refresh of statistics.

Refer to the *Cisco ONS 15454 Troubleshooting Guide* for definitions of the SONET POS parameters. CTC displays a different set of parameters for high-level data link control (HDLC) and frame-mapped generic framing procedure (GFP-F) framing modes.

Displaying ML-Series Ethernet Statistics in CTC

The Ethernet statistics window lists Ethernet port-level statistics. It is similar in appearance to the POS statistics window with different statistic parameters. The ML-Series Ethernet ports are zero based. Display the CTC card view for the ML-Series card and click the **Performance > Ether Ports** tabs to display the window. Table 2-2 describes the buttons in the EtherPorts window.

Table 2-2 ML-Series Ethernet Statistics Fields and Buttons

Button	Description
Refresh	Queries the current values from the card and updates the CTC display.
Baseline	Resets the software counters (in that particular CTC client only) temporarily to zero without affecting the actual statistics on the card. From that point on, only counters displaying the change from the temporary baseline are displayed by this CTC client. These new baselined counters appear as long as the user displays the Performance window. If the user navigates to another CTC window and comes back to the Performance window, the true actual statistics retained by the card are shown.
Auto-Refresh	Sets a time interval for the automatic refresh of statistics.

Refer to the *Cisco ONS 15454 Troubleshooting Guide* for definitions of the Ethernet parameters. CTC displays a different set of parameters for HDLC and GFP-F framing modes.

Displaying ML-Series Ethernet Ports Provisioning Information on CTC

The Ethernet port provisioning window displays the provisioning status of the Ethernet ports. Click the **Provisioning > Ether Ports** tabs to display this window. For ML-Series cards, the user must configure ML-Series Ethernet ports and POS ports using the Cisco IOS CLI.

The following fields can be provisioned using CTC: Port Name, Pre-Service Alarm Suppression (PSAS), and Soak Time. Click the Port Name field to assign a name to the port. For more information on provisioning these fields, refer to the "Change Card Settings" chapter in the *Cisco ONS 15454 Procedure Guide*.

Note The port name can also be configured in Cisco IOS. The port name field configured in CTC and the port name configured in Cisco IOS are independent of each other, and will not match unless the same name is used to configure the port name in both CTC and Cisco IOS.

The Provisioning > Ether Ports tab displays the following information:

- Port #—The fixed number identifier for the specific port.
- Port Name—Configurable 12-character alphanumeric identifier for the port.
- Admin State—Configured port state, which is administratively active or inactive. Possible values are UP and DOWN.
- PSAS—A check indicates alarm suppression is set on the port for the time designated in the Soak Time column.
- Soak Time—Desired soak time in hours and minutes. Use this column when you have checked PSAS to suppress alarms. Once the port detects a signal, the countdown begins for the designated soak time. Soak time hours can be set from 0 to 48. Soak time minutes can be set from 0 to 45 in 15 minute increments.
- Link State—Status between signaling points at port and attached device. Possible values are UP and DOWN.
- Operating Speed—ML-100T-8 possible values are Auto, 10Mbps, or 100Mbps.
- Operating Duplex—Setting of the port. ML-100T-8 possible values are Auto, Full, or Half.
- Flow Control—Negotiated flow control mode. ML-100T-8 possible values are None or Symmetrical.

Note Auto indicates the port is set to autonegotiate capabilities with the attached link partner.

Displaying ML-Series POS Ports Provisioning Information on CTC

The POS ports provisioning window displays the provisioning status of the card's POS ports. Click the **Provisioning > POS Ports** tabs to display this window. For ML-Series cards, the user must configure ML-Series Ethernet ports and POS ports using the Cisco IOS CLI.

The following fields can be provisioned using CTC: Port Name, PSAS, and Soak Time. Click in the Port Name field to assign a name to the port. For more information on provisioning these fields, refer to the "Change Card Settings" chapter in the *Cisco ONS 15454 Procedure Guide*.

Note The port name can also be configured in Cisco IOS. The port name field configured in CTC and the port name configured in Cisco IOS are independent of each other and will not match unless the same name is used to configure the port name in both CTC and Cisco IOS.

The Provisioning > POS Ports tab displays the following information:

- Port #—Fixed number identifier for the specific port.
- Port Name—Configurable 12-character alphanumeric identifier for the port.
- Admin State—Configured administrative port state, which is active or inactive. Possible values are UP and DOWN. For the UP value to appear, a POS port must be both administratively active and have a SONET/SDH circuit provisioned.
- PSAS—A check indicates alarm suppression is set on the port for the time designated in the Soak Time column.
- Soak Time—Desired soak time in hours and minutes. Use this column when you have checked PSAS to suppress alarms. Once the port detects a signal, the countdown begins for the designated soak time. Soak time hours can be set from 0 to 48. Soak time minutes can be set from 0 to 45 in 15 minute increments.
- MTU—The maximum transfer unit, which is the largest acceptable packet size for that port. This value cannot be configured on the Cisco ONS 15310-CL and the Cisco ONS 15310-MA ML-Series card.
- Link State—Status between signaling points at the port and an attached device. Possible values are UP and DOWN.
- Framing Type- HDLC or frame-mapped generic framing procedure (GFP-F) framing type shows the POS framing mechanism being employed on the port

Displaying SONET Alarms

To view SONET alarms on the ML-Series card, click the **Alarms** tab**.**

CTC manages the ML-Series card SONET alarm behavior in the same manner as it manages alarm behavior for other Cisco ONS 15310-CL and the Cisco ONS 15310-MA SONET traffic. Click the **Provisioning > Alarm Profiles** tabs for the Ethernet and POS port alarm profile information. Refer to the *Cisco ONS 15454 Troubleshooting Guide* for detailed information.

Displaying J1 Path Trace

The J1 Path Trace is a repeated, fixed-length string comprised of 64 consecutive J1 bytes. You can use the string to monitor interruptions or changes to SONET circuit traffic. Click the **Maintenance > Path Trace** tabs for the J1 Path Trace information.

For information on J1 Path Trace, refer to the *Cisco ONS 15454 Troubleshooting Guide.*

Provisioning SONET Circuits

CTC provisions and edits STS level circuits for the two POS ports of the ML-Series card in the same manner as it provisions other Cisco ONS 15310-CL and Cisco ONS 15310-MA SONET OC-N cards. The ONS 15310-CL ML-Series card supports both contiguous concatenation (CCAT) and virtual concatenation (VCAT) circuits. Refer to the “Create Circuits” chapter of the *Cisco ONS 15454 Procedure Guide* to create SONET STS circuits.

Note The initial state of the ML-Series card POS port is inactive. A Cisco IOS POS interface command of **no shutdown** is required to carry traffic on the SONET circuit.

CHAPTER 3

Initial Configuration of the ML-Series Card

This chapter describes the initial configuration of the ML-Series card and contains the following major sections:

Hardware Installation

This section lists hardware installation tasks, including booting up the ML-Series card. Because the ONS 15310 card slots can be preprovisioned for an ML-Series line card, the following physical operations can be performed before or after the provisioning of the slot has taken place.

1. Install the ML-Series card into the ONS 15310. For physical installation instructions, refer to the *Cisco ONS 15454 Troubleshooting Guide*.
2. Connect the Ethernet cables to the ML-Series card.
3. Connect the console terminal to the ML-Series card (optional).

Note A NO-CONFIG condition is reported in CTC under the Alarms pane when an ML-Series card is inserted and no valid Cisco IOS startup configuration file exists. Loading or creating this file clears the condition. See the "Startup Configuration File" section on page 3-5 for information on loading or creating the file.

Cisco IOS on the ML-Series Card

The Cisco IOS software image used by the ML-Series card is not permanently stored on the ML-Series card but in the flash memory of the 15310-CL-CTX or CTX2500 card. During a hard reset, the Cisco IOS software image is downloaded from the flash memory of the 15310-CL-CTX or CTX2500 to the memory cache of the ML-Series card. The cached image is then decompressed and initialized for use by the ML-Series card.

During a soft reset, which reloads or warm restarts the ML-Series card, the ML-Series card checks the cache for a Cisco IOS image. If a valid and current Cisco IOS image exists, the ML-Series card decompresses and initializes the image. If the image does not exist, the ML-Series requests a new copy of the Cisco IOS image from the 15310-CL-CTX or CTX2500. Caching the Cisco IOS image provides a significant time savings when a soft reset is performed.

To use CTC to reset the ML-Series card with a hard reset or soft reset, at the CTC card-level view or node-level view, right-click on the ML-Series card and click **Hard-reset Card** or **Soft-reset Card**. A hard reset also requires that the ML-Series card is in the out of service (OOS) state, which is set under the Inventory tab. Then click **Yes** at the confirmation dialog that appears. You can also initiate a hard reset by removing and reinserting the ML-Series card. You can initiate a soft reset through Cisco IOS with the privileged EXEC **reboot** command. For TL1 commands, refer to the *Cisco ONS SONET TL1 Command Guide*.

Caution A soft reset or a hard reset on the Cisco ONS 15310 ML-Series card is service-affecting.

There are four ways to access the ML-Series card Cisco IOS configuration. The two out-of-band options are opening a Cisco IOS session on CTC and telnetting to the node IP Address and 2001. The two-in-band signalling options are telnetting to a configured management interface and directly connecting to the console port.

Opening a Cisco IOS Session Using CTC

Users can initiate a Cisco IOS CLI session for the ML-Series card using CTC. Click the **IOS** tab at the card-level CTC view, then click the **Open IOS Command Line Interface (CLI)** button. A window opens and a standard Cisco IOS CLI User EXEC command mode prompt appears.

Note A Cisco IOS startup configuration file must be loaded and the ML-Series card must be installed and initialized prior to opening a Cisco IOS CLI session on CTC. See the "Startup Configuration File" section on page 3-5 for more information.

Telnetting to the Node IP Address and Slot Number

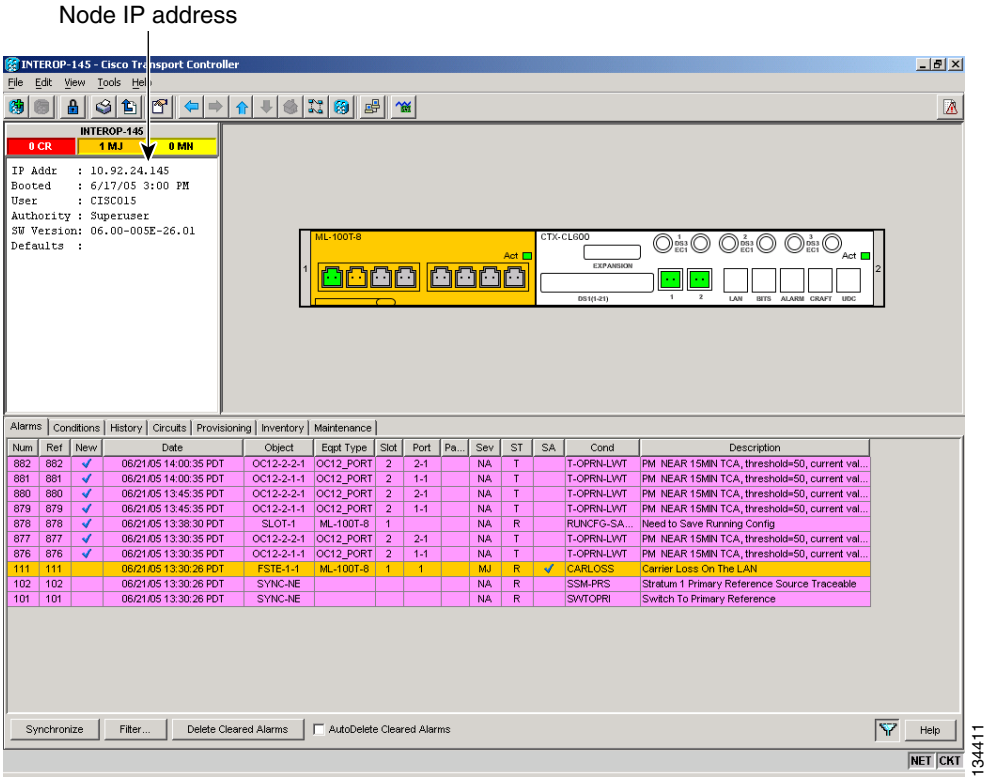
Users can telnet to the Cisco IOS CLI using the IP address and the port number (2000 plus the slot number).

Note A Cisco IOS startup configuration file must be loaded and the ML-Series card must be installed and initialized prior to telnetting to the ML-Series card. See the "Startup Configuration File" section on page 3-5 for more information.

Note If the ONS 15310 node is set up as a proxy server, where one ONS 15310 node in the ring acts as a gateway network element (GNE) for the other nodes in the ring, telnetting over the GNE firewall to the IP address and slot number of a non-GNE or end network element (ENE) requires the user's Telnet client to be SOCKS v5 aware (RFC 1928). Configure the Telnet client to recognize the GNE as the SOCKS v5 proxy for the Telnet session and to recognize the ENE as the host.

Step 1 Obtain the node IP address from the IP Addr field shown at the CTC node view (Figure 3-1).

Figure 3-1 CTC Node View Showing IP Address



Step 2 If you are telnetting into an ONS 15310-CL with an ML-Series card, use the IP address and the port number 2001 as the Telnet address in your preferred communication program. For example with the IP address of 10.92.18.124 on the ONS 15310-CL in the example, you would enter or telnet `10.92.18.124 2001`. The slot number is always 1 for the ONS 15310-CL.

Step 3 If you are telnetting into an ONS 15310-MA with an ML-Series card, use the IP address and the port number (2000 plus the slot number) as the Telnet address in your preferred communication program. For example, with an IP address of 10.92.18.125 on an ONS 15310-CL with an ML-Series card in slot 5, you would enter or telnet to `10.92.18.125 2005.`.

Telnetting to a Management Port

Users can access the ML-Series through a standard Cisco IOS management port in the same manner as other Cisco IOS platforms. For further details about configuring ports and lines for management access, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide*.

As a security measure, the vty lines used for Telnet access are not fully configured. In order to gain Telnet access to the ML-Series card, you must configure the vty lines via the serial console connection or preload a startup-configuration file that configures the vty lines. A port on the ML-Series must first be configured as the management port; see the "Configuring the Management Port" section on page 3-6 or the "Loading a Cisco IOS Startup Configuration File Through CTC" section on page 3-8.

ML-Series IOS CLI Console Port

The ML-Series card has an RJ-11 serial console port on the card faceplate labeled Console. It enables communication from the serial port of a PC or workstation running terminal emulation software to the Cisco IOS CLI on a specific ML-Series card.

RJ-11 to RJ-45 Console Cable Adapter

Due to space limitations on the ML-Series card faceplate, the console port is an RJ-11 modular jack instead of the more common RJ-45 modular jack. Cisco supplies an RJ-11 to RJ-45 console cable adapter with each ML-Series card. After connecting the adapter, the console port functions like the standard Cisco RJ-45 console port. Figure 3-2 shows the RJ-11 to RJ-45 console cable adapter.

Figure 3-2 Console Cable Adapter

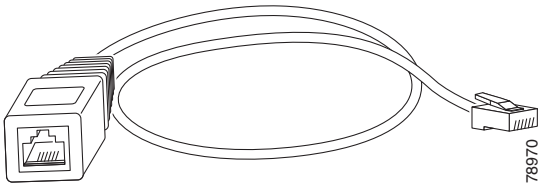


Table 3-1 shows the mapping of the RJ-11 pins to the RJ-45 pins.

Table 3-1 RJ-11 to RJ-45 Pin Mapping

RJ-11 Pin	RJ-45 Pin
1	1
2	2
3	3
4	4
None	5
5	6
None	7
6	8

Connecting a PC or Terminal to the Console Port

Use the supplied cable, an RJ-11 to RJ-45 console cable adapter, and a DB-9 adapter to connect a PC to the ML-Series console port.

The PC must support VT100 terminal emulation. The terminal-emulation software—frequently a PC application such as HyperTerminal or Procomm Plus—makes communication between the ML-Series and your PC or terminal possible during the setup program.

Step 1 Configure the data rate and character format of the PC or terminal to match these console port default settings:

- 9600 baud

- 8 data bits
- 1 stop bit
- No parity

Step 2 Insert the RJ-45 connector of the supplied cable into the female end of the supplied console cable adapter.

Step 3 Insert the RJ-11 modular plug end of the supplied console cable adapter into the RJ-11 serial console port, labeled CONSOLE, on the ML-Series card faceplate.

Step 4 Attach the supplied RJ-45-to-DB-9 female DTE adapter to the nine-pin DB-9 serial port on the PC.

Step 5 Insert the other end of the supplied cable in the attached adapter.

Startup Configuration File

The ML-Series card needs a startup configuration file in order to configure itself beyond the default configuration when it resets. If no startup configuration file exists in the 15310-CL-CTX or the CTX 2500 flash memory, then the card boots up to a default configuration. Users can manually set up the startup configuration file through the serial console port and the Cisco IOS CLI configuration mode or load a Cisco IOS supplied sample startup configuration file through CTC. A running configuration becomes a startup configuration file when saved with a **copy running-config startup-config** command.

It is not possible to establish a Telnet connection to the ML-Series card until a startup configuration file is loaded onto the ML-Series card. Access is available through the console port.

Caution The **copy running-config startup-config** command saves a startup configuration file to the flash memory of the ML-Series card. This operation is confirmed by the appearance of the text "[OK]" in the Cisco IOS CLI session. The startup configuration file is also saved to the ONS node's database restoration file after approximately 30 additional seconds.

Caution Accessing the read-only memory monitor mode (ROMMON) on the ML-Series card without the assistance of Cisco personnel is not recommended. This mode allows actions that can render the ML-Series card inoperable. The ML-Series card ROMMON is preconfigured to boot the correct Cisco IOS software image for the ML-Series card.

Caution The maximum permitted size of the startup configuration file on the ONS 15310 ML-Series card is 96 kilobytes.

Note When the running configuration file is altered, a RUNCFG-SAVENEED condition appears in CTC. This condition is a reminder to enter a **copy running-config startup-config** command in the Cisco IOS CLI, or configuration changes will be lost when the ML-Series card reboots.

Manually Creating a Startup Configuration File Through the Serial Console Port

Configuration through the serial console port is familiar to those who have worked with other products using Cisco IOS. At the end of the configuration procedure, the **copy running-config startup-config** command saves a startup configuration file.

The serial console port gives the user visibility to the entire booting process of the ML-Series card. During initialization the ML-Series card first checks for a locally, valid cached copy of Cisco IOS. It then either downloads the Cisco IOS software image from the 15310-CL-CTX or the CTX 2500 or proceeds directly to decompressing and initializing the image. Following Cisco IOS initialization the CLI prompt appears, at which time the user can enter the Cisco IOS CLI configuration mode and setup the basic ML-Series configuration.

Passwords

There are two types of passwords that you can configure for an ML-Series card: an enable password and an enable secret password. For maximum security, make the enable password different from the enable secret password.

- Enable password—The enable password is an unencrypted password. It can contain any number of uppercase and lowercase alphanumeric characters. Give the enable password only to users permitted to make configuration changes to the ML-Series card.
- Enable secret password—The enable secret password is a secure, encrypted password. By setting an encrypted password, you can prevent unauthorized configuration changes. On systems running Cisco IOS software, you must enter the enable secret password before you can access global configuration mode.

  An enable secret password can contain from 1 to 25 uppercase and lowercase alphanumeric characters. The first character cannot be a number. Spaces are valid password characters. Leading spaces are ignored; trailing spaces are recognized.

Passwords are configured in the "Configuring the Management Port" section on page 3-6.

Configuring the Management Port

Because there is no separate management port on ML-Series cards, any Fast Ethernet interface (0-7), or any POS interface (0-1) can be configured as a management port.

You can remotely configure the ML-Series card through the management port, but first you must configure an IP address so that the ML-Series card is reachable or load a startup configuration file. You can manually configure the management port interface from the Cisco IOS CLI via the serial console connection.

To configure Telnet for remote management access, perform the following procedure, beginning in user EXEC mode:

	Command	Purpose
Step 1	`Router> enable`	Activates user EXEC (or enable) mode. The # prompt indicates enable mode.
Step 2	`Router# configure terminal`	Activates global configuration mode. You can abbreviate the command to **config t**. The Router(config)# prompt indicates that you are in global configuration mode.

	Command	Purpose
Step 3	`Router(config)# enable password password`	Sets the enable password. See the "Passwords" section on page 3-6.
Step 4	`Router(config)# enable secret password`	Allows you to enter an enable secret password. See the "Passwords" section on page 3-6. A user must enter the enable secret password to gain access to global configuration mode.
Step 5	`Router(config)# interface type number` `Router(config-if)#`	Activates interface configuration mode on the interface.
Step 6	`Router(config-if)# ip address ip-address subnetmask`	Allows you to enter the IP address and IP subnet mask for the interface specified in Step 5.
Step 7	`Router(config-if)# no shutdown`	Enables the interface.
Step 8	`Router(config-if)# exit` `Router(config)#`	Returns to global configuration mode.
Step 9	`Router(config)# line vty line-number`	Activates line configuration mode for virtual terminal connections. Commands entered in this mode control the operation of Telnet sessions to the ML-Series card.
Step 10	`Router(config-line)# password password`	Allows you to enter a password for Telnet sessions.
Step 11	`Router(config-line)# end` `Router#`	Returns to privileged EXEC mode.
Step 12	`Router# copy running-config startup-config`	(Optional) Saves your configuration changes to NVRAM.

After you have completed configuring remote management on the management port, you can use Telnet to remotely assign and verify configurations.

Configuring the Hostname

In addition to the system passwords and enable password, your initial configuration should include a hostname to easily identify your ML-Series card. To configure the hostname, perform the following task, beginning in enable mode:

	Command	Purpose
Step 1	`Router# configure terminal`	Activates global configuration mode.
Step 2	`Router(config)# hostname name-string`	Allows you to enter a system name. In this example, we set the hostname to "Router."
Step 3	`Router(config)# end`	Returns to privileged EXEC mode.
Step 4	`Router# copy running-config startup-config`	(Optional) Copies your configuration changes to NVRAM.

Loading a Cisco IOS Startup Configuration File Through CTC

CTC allows a user to load the startup configuration file required by the ML-Series card. A Cisco-supplied sample Cisco IOS startup configuration file, named **Basic-IOS-startup-config.txt**, is available on the Cisco ONS 15310 software CD. CISCO15 is the Cisco IOS CLI default line password and the enable password for this configuration. Users can also create their own startup configuration file (see the "Manually Creating a Startup Configuration File Through the Serial Console Port" section on page 3-6).

CTC can load a Cisco IOS startup configuration file into the 15310-CL-CTX or CTX 2500 card flash before the ML-Series card is physically installed in the slot. When installed, the ML-Series card downloads and applies the Cisco IOS software image and the preloaded Cisco IOS startup-configuration file. Preloading the startup configuration file allows an ML-Series card to immediately operate as a fully configured card when inserted into the ONS 15310.

If the ML-Series card is booted up prior to the loading of the Cisco IOS startup configuration file into 15310-CL-CTX or CTX 2500 card flash, then the ML-Series card must be reset to use the Cisco IOS startup configuration file or the user can issue the command **copy start run** at the Cisco IOS CLI to configure the ML-Series card to use the Cisco IOS startup configuration file.

This procedure details the initial loading of a Cisco IOS Startup Configuration file through CTC.

Step 1 At the card-level view of the ML-Series card, click the **IOS** tab (Figure 3-1 on page 3-3).

The CTC IOS window appears.

Step 2 Click the **IOS startup config** button.

The config file dialog box appears.

Step 3 Click the **Local -> CTX** button.

Step 4 The sample Cisco IOS startup configuration file can be installed from either the ONS 15310 software CD or from a PC or network folder:

- To install the Cisco supplied startup config file from the ONS 15310 software CD, insert the CD into the CD drive of the PC or workstation. Using the CTC config file dialog box, navigate to the CD drive of the PC or workstation, and double-click the **Basic-IOS-startup-config.txt** file.
- To install the Cisco supplied config file from a PC or network folder, navigate to the folder containing the desired Cisco IOS startup config file and double-click the desired Cisco IOS startup config file.

Step 5 At the Are you sure? dialog box, click the **Yes** button.

The Directory and Filename fields on the configuration file dialog update to reflect that the Cisco IOS startup config file is loaded onto the 15310-CL-CTX.

Step 6 Load the Cisco IOS startup config file from the 15310-CL-CTX to the ML-Series card:

a. If the ML-Series card has already been installed, right-click on the ML-Series card at the node-level or card-level CTC view and select **Soft-reset**.

After the reset, the ML-Series card runs under the newly loaded Cisco IOS startup configuration.

b. If the ML-Series card is not yet installed, installing the ML-Series card into the slot loads and runs the newly loaded Cisco IOS startup configuration on the ML-Series card.

Caution A soft reset or a hard reset on the ONS 15310 ML-Series card is service-affecting.

Note If there is a parsing error when the Cisco IOS startup configuration file is downloaded and parsed at initialization, an ERROR-CONFIG alarm is reported and appears under the CTC alarms tab or in TL1. No other Cisco IOS error messages regarding the parsing of text are reported to the CTC or in TL1. An experienced Cisco IOS user can locate and troubleshoot the line in the startup configuration file that produced the parsing error by opening the Cisco IOS CLI and entering a **copy start run** command.

Note A standard ONS 15310 database restore reinstalls the Cisco IOS startup config file, but does not implement the Cisco IOS startup config on the ML-Series. Complete Step 6 to load the Cisco IOS startup config file from the 15310-CL-CTX to the ML-Series card.

Database Restore of the Startup Configuration File

The ONS 15310-CL includes a database restoration feature. Restoring the database will reconfigure a node and the installed line cards to the saved provisioning, except for the ML-Series card. The ML-Series card does not automatically restore the startup configuration file saved in the database.

A user can load the saved startup configuration file onto the ML-Series card in two ways. He can revert completely to the saved startup configuration and lose any additional provisioning in the unsaved running configuration, which is a restoration scheme similar to other ONS cards, or he can install the saved startup configuration file on top of the current running configuration, which is a merging restoration scheme used by many Cisco Catalyst devices.

To revert completely to the startup configuration file saved in the restored database, the user needs to soft reset the ML-Series card. Right-click the ML-Series card in CTC and choose **Soft-reset** or use the Cisco IOS CLI **reload** command to reset the ML-Series card.

To merge the saved startup configuration file with the running configuration, use the Cisco IOS CLI **copy startup-config running-config** command. This restoration scheme should only be used by experienced users with an understanding of the current running configuration and the Cisco IOS **copy** command. The **copy startup-config running-config** command will not reset the ML-Series card. The user also needs to use the Cisco IOS CLI **copy running-config startup-config** command to save the new merged running configuration to the startup configuration file.

Cisco IOS Command Modes

The Cisco IOS user interface has several different modes. The commands available to you depend on which mode you are in. To get a list of the commands available in a given mode, type a question mark (?) at the system prompt.

Table 3-2 describes the most commonly used modes, how to enter the modes, and the resulting system prompts. The system prompt helps you identify which mode you are in and, therefore, which commands are available to you.

Note When a process makes unusually heavy demands on the CPU of the ML-Series card, it might impair CPU response time and cause a CPUHOG error message to appear on the console. This message indicates which process used a large number of CPU cycles, such as the updating of the routing table with a large number of routes due to an event. Seeing this message as a result of card reset or other infrequent events should not be a cause for concern.

Table 3-2 Cisco IOS Command Modes

Mode	What You Use It For	How to Access	Prompt
User EXEC	Connect to remote devices, change terminal settings on a temporary basis, perform basic tests, and display system information.	Log in.	`Router>`
Privileged EXEC (also called Enable mode)	Set operating parameters. The privileged command set includes the commands in user EXEC mode, as well as the **configure** command. Use this command mode to access the other command modes.	From user EXEC mode, enter the **enable** command and the enable password.	`Router#`
Global configuration	Configure features that affect the system as a whole.	From privileged EXEC mode, enter the **configure terminal** command.	`Router(config)#`
Interface configuration	Enable features for a particular interface. Interface commands enable or modify the operation of a Fast Ethernet or POS port.	From global configuration mode, enter the **interface** *type number* command. For example, enter **interface fastethernet 0** for Fast Ethernet or **interface pos 0** for POS interfaces.	`Router(config-if)#`
Line configuration	Configure the console port or vty line from the directly connected console or the virtual terminal used with Telnet.	From global configuration mode, enter the **line console 0** command to configure the console port or the **line vty** *line-number* command to configure a vty line.	`Router(config-line)#`

When you start a session on the ML-Series card, you begin in user EXEC mode. Only a small subset of the commands are available in user EXEC mode. To have access to all commands, you must enter privileged EXEC mode, also called Enable mode. From privileged EXEC mode, you can type in any EXEC command or access global configuration mode. Most of the EXEC commands are single-use commands, such as **show** commands, which show the current configuration status, and **clear** commands, which clear counters or interfaces. The EXEC commands are not saved across reboots of the ML-Series card.

The configuration modes allow you to make changes to the running configuration. If you later save the configuration, these commands are stored across ML-Series card reboots. You must start in global configuration mode. From global configuration mode, you can enter interface configuration mode, subinterface configuration mode, and a variety of protocol-specific modes.

ROMMON mode is a separate mode used when the ML-Series card cannot boot properly. For example, your ML-Series card might enter ROM monitor mode if it does not find a valid system image when it is booting, or if its configuration file is corrupted at startup.

Using the Command Modes

The Cisco IOS command interpreter, called the EXEC, interprets and executes the commands you enter. You can abbreviate commands and keywords by entering just enough characters to make the command unique from other commands. For example, you can abbreviate the **show** command to **sh** and the **configure terminal** command to **config t**.

Exit

When you type **exit**, the ML-Series card backs out one level. In general, typing **exit** returns you to global configuration mode. Enter **end** to exit configuration mode completely and return to privileged EXEC mode.

Getting Help

In any command mode, you can get a list of available commands by entering a question mark (?).

```
Router> ?
```

To obtain a list of commands that begin with a particular character sequence, type in those characters followed immediately by the question mark (?). Do not include a space. This form of help is called word help, because it completes a word for you.

```
Router# co?
configure
```

To list keywords or arguments, enter a question mark in place of a keyword or argument. Include a space before the question mark. This form of help is called command syntax help, because it reminds you which keywords or arguments are applicable based on the command, keywords, and arguments you have already entered.

```
Router# configure ?
  memory              Configure from NV memory
  network             Configure from a TFTP network host
  overwrite-network   Overwrite NV memory from TFTP network host
  terminal            Configure from the terminal
  <cr>
```

To redisplay a command you previously entered, press the Up Arrow key. You can continue to press the Up Arrow key to see more of the previously issued commands.

Tip If you are having trouble entering a command, check the system prompt, and enter the question mark (?) for a list of available commands. You might be in the wrong command mode or using incorrect syntax.

You can press **Ctrl-Z** or type **end** in any mode to immediately return to privileged EXEC (enable) mode, instead of entering **exit**, which returns you to the previous mode.

CHAPTER 4

Configuring Interfaces on the ML-Series Card

This chapter describes basic interface configuration for the ML-Series card to help you get your ML-Series card up and running. Advanced packet-over-SONET (POS) interface configuration is covered in Chapter 5, "Configuring POS on the ML-Series Card." For more information about the Cisco IOS commands used in this chapter, refer to the *Cisco IOS Command Reference* publication.

This chapter contains the following major sections:

General Interface Guidelines

The main function of the ML-Series card is to relay packets from one data link to another. Consequently, you must configure the characteristics of the interfaces, which receive and send packets. Interface characteristics include, but are not limited to, IP address, address of the port, data encapsulation method, and media type.

Many features are enabled on a per-interface basis. Interface configuration mode contains commands that modify the interface operation (for example, of an Ethernet port). When you enter the **interface** command, you must specify the interface type and number.

The following general guidelines apply to all physical and virtual interface configuration processes:

- All interfaces have a name that is composed of an interface type (word) and a Port ID (number). For example, Fast Ethernet 2.
- Configure each interface with a bridge-group or IP address and IP subnet mask.
- VLANs are supported through the use of subinterfaces. The subinterface is a logical interface configured separately from the associated physical interface.
- Each physical interface, including the internal POS interfaces, has an assigned MAC address.

MAC Addresses

Every port or device that connects to an Ethernet network needs a MAC address. Other devices in the network use MAC addresses to locate specific ports in the network and to create and update routing tables and data structures.

To find MAC addresses for a device, use the **show interfaces** command, as follows:

```
ML_Series# show interfaces fastethernet 0
FastEthernet0 is up, line protocol is up
  Hardware is epif_port, address is 000b.fcfa.339e (bia 000b.fcfa.339e)
  Description: 100 mbps full duplex q-in-q tunnel
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
     reliability 255/255, txload 18/255, rxload 200/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, 100Mb/s, 100BaseTX
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:00, output 00:00:00, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: weighted fair
  Output queue: 0/1000/64/0 (size/max total/threshold/drops)
     Conversations  0/0/256 (active/max active/max total)
     Reserved Conversations 0/0 (allocated/max allocated)
     Available Bandwidth 75000 kilobits/sec
  30 second input rate 78525000 bits/sec, 144348 packets/sec
  30 second output rate 7363000 bits/sec, 13537 packets/sec
     4095063706 packets input, 3885007012 bytes
     Received 0 broadcasts (0 IP multicast)
     2 runts, 0 giants, 0 throttles
     4 input errors, 0 CRC, 0 frame, 1 overrun, 0 ignored
     0 watchdog, 0 multicast
     0 input packets with dribble condition detected
     1463732665 packets output, 749573412 bytes, 0 underruns
     131072 output errors, 131072 collisions, 0 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
```

Interface Port ID

The interface port ID designates the physical location of the interface within the ML-Series card. It is the name that you use to identify the interface you are configuring. The system software uses interface port IDs to control activity within the ML-Series card and to display status information. Interface port IDs are not used by other devices in the network; they are specific to the individual ML-Series card and its internal components and software.

The ML-100T-8 port IDs for the eight Fast Ethernet interfaces are Fast Ethernet 0 through 7. The ML-Series card features two POS ports. The ML-Series port IDs for the two POS interfaces are POS 0 and 1. You can use user-defined abbreviations such as f0 through f7 to configure the eight Fast Ethernet interfaces, and POS0 and POS1 to configure the two POS ports.

You can use Cisco IOS **show** commands to display information about any or all the interfaces of the ML-Series card.

Basic Interface Configuration

The following general configuration instructions apply to all interfaces. Before you configure interfaces, develop a plan for a bridge or routed network.

To configure an interface, do the following:

Step 1 Enter the **configure** EXEC command at the privileged EXEC prompt to enter global configuration mode. The key word *your-password* is the password set up by the user in the initial configuration of the ML-Series card.

```
ML_Series> enable
Password:<your-password>
ML_Series# configure terminal
ML_Series(config)#
```

Step 2 Enter the **interface** command, followed by the interface type (for example, fastethernet or pos) and its interface port ID (see the "Interface Port ID" section on page 4-2).

For example, to configure a Fast Ethernet port, enter this command:

```
ML_Series(config)# interface fastethernet number
```

Step 3 Follow each **interface** command with the interface configuration commands required for your particular interface.

The commands you enter define the protocols and applications that will run on the interface. The ML-Series card collects and applies commands to the **interface** command until you enter another **interface** command or a command that is not an interface configuration command. You can also enter **end** to return to privileged EXEC mode.

Step 4 Check the status of the configured interface by entering the EXEC **show interface** command.

```
ML_Series# show interfaces fastethernet 0
FastEthernet0 is up, line protocol is up
  Hardware is epif_port, address is 000b.fcfa.339e (bia 000b.fcfa.339e)
  Description: 100 mbps full duplex q-in-q tunnel
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
     reliability 255/255, txload 18/255, rxload 200/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, 100Mb/s, 100BaseTX
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:00, output 00:00:00, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: weighted fair
  Output queue: 0/1000/64/0 (size/max total/threshold/drops)
     Conversations  0/0/256 (active/max active/max total)
     Reserved Conversations 0/0 (allocated/max allocated)
     Available Bandwidth 75000 kilobits/sec
  30 second input rate 78525000 bits/sec, 144348 packets/sec
  30 second output rate 7363000 bits/sec, 13537 packets/sec
     4095063706 packets input, 3885007012 bytes
     Received 0 broadcasts (0 IP multicast)
     2 runts, 0 giants, 0 throttles
     4 input errors, 0 CRC, 0 frame, 1 overrun, 0 ignored
     0 watchdog, 0 multicast
     0 input packets with dribble condition detected
     1463732665 packets output, 749573412 bytes, 0 underruns
     131072 output errors, 131072 collisions, 0 interface resets
     0 babbles, 0 late collision, 0 deferred
```

```
0 lost carrier, 0 no carrier
0 output buffer failures, 0 output buffers swapped out
```

Basic Fast Ethernet and POS Interface Configuration

ML-Series cards support Fast Ethernet and POS interfaces. This section provides some examples of configurations for all interface types.

To configure an IP address or bridge-group number on a Fast Ethernet or POS interface, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	`ML_Series(config)# interface` *type number*	Activates interface configuration mode to configure either the Fast Ethernet interface or the POS interface.
Step 2	`ML_Series(config-if)# {ip address` *ip-address subnet-mask* `| bridge-group` *bridge-group-number*`}`	Sets the IP address and IP subnet mask to be assigned to the interface. or Assigns a network interface to a bridge group.
Step 3	`ML_Series(config-if)# no shutdown`	Enables the interface by preventing it from shutting down.
Step 4	`ML_Series(config)# end`	Returns to privileged EXEC mode.
Step 5	`ML_Series# copy running-config startup-config`	(Optional) Saves configuration changes to flash database.

Configuring the Fast Ethernet Interfaces

To configure the IP address or bridge-group number, autonegotiation, and flow control on a Fast Ethernet interface, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	`ML_Series(config)# interface fastethernet` *number*	Activates interface configuration mode to configure the Fast Ethernet interface.
Step 2	`ML_Series(config-if)# {ip address` *ip-address subnet-mask* `| bridge-group` *bridge-group-number*`}`	Sets the IP address and IP subnet mask to be assigned to the interface. or Assigns a network interface to a bridge group.
Step 3	`ML_Series(config-if)# [no] speed {10 | 100 | auto}`	Configures the transmission speed for 10 or 100 Mbps. If you set the speed or duplex for **auto**, you enable autonegotiation on the system—the ML-Series card matches the speed and duplex mode of the partner node.

	Command	Purpose
Step 4	`ML_Series(config-if)# [no] duplex {full | half | auto}`	Sets full duplex, half duplex, or autonegotiate mode.
Step 5	`ML_Series(config-if)# flowcontrol send {on | off | desired}`	(Optional) Sets the send flow control value for an interface. Flow control works only with port-level policing. ML-Series card Fast Ethernet port flow control is IEEE 802.3x compliant.
Step 6	`ML_Series(config-if)# no shutdown`	Enables the interface by preventing it from shutting down.
Step 7	`ML_Series(config)# end`	Returns to privileged EXEC mode.
Step 8	`ML_Series# copy running-config startup-config`	(Optional) Saves your configuration changes to the flash database.

Example 4-1 shows how to do the initial configuration of a Fast Ethernet interface with an IP address, autonegotiated speed, and autonegotiated duplex.

Example 4-1 Initial Configuration of a Fast Ethernet Interface

```
ML_Series(config)# interface fastethernet 1
ML_Series(config-if)# ip address 10.1.2.4 255.0.0.0
ML_Series(config-if)# speed auto
ML_Series(config-if)# duplex auto
ML_Series(config-if)# no shutdown
ML_Series(config-if)# end
ML_Series# copy running-config startup-config
```

Configuring the POS Interfaces

Encapsulation changes on POS ports are allowed only when the interface is in a manual shutdown (ADMIN_DOWN). For advanced POS interface configuration, see Chapter 5, "Configuring POS on the ML-Series Card."

Note The initial state of the ONS 15310-CL and ONS 15310-MA ML-Series card POS port is inactive. A POS interface command of **no shutdown** is required to carry traffic on the SONET circuit.

To configure the IP address, bridge group, or encapsulation for the POS interface, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	`ML_Series(config)# interface pos` *number*	Activates interface configuration mode to configure the POS interface.
Step 2	`ML_Series(config-if)# {ip address` *ip-address subnet-mask* `| bridge-group` *bridge-group-number*`}`	Sets the IP address and subnet mask. or Assigns a network interface to a bridge group.

	Command	Purpose
Step 3	`ML_Series(config-if)# shutdown`	Manually shuts down the interface. Encapsulation changes on POS ports are allowed only when the interface is shut down (ADMIN_DOWN).
Step 4	`ML_Series(config-if)# encapsulation` *type*	Sets the encapsulation type. Valid values are: • **hdlc**—Cisco high-level data link control (HDLC) • **lex**—(Default) LAN extension, special encapsulation for use with Cisco ONS Ethernet line cards • **ppp**—Point-to-Point Protocol **Note** Under GFP-F framing, the ONS 15310-CLand ONS 15310-MA ML-Series card is restricted to LEX encapsulation.
Step 5	`ML_Series(config-if)# no shutdown`	Restarts the shutdown interface.
Step 6	`ML_Series(config)# end`	Returns to privileged EXEC mode.
Step 7	`ML_Series# copy running-config startup-config`	(Optional) Saves configuration changes to NVRAM.

Monitoring Operations on the Fast Ethernet Interfaces

To verify the settings after you have configured the interfaces, enter the **show interface** command. For additional information on monitoring the operations on POS interfaces, see the "Configuring POS on the ML-Series Card" chapter.

Example 4-2 shows the output from the **show interface** command, which displays the status of the interface including port speed and duplex operation.

Example 4-2 show interface Command Output

```
ML_Series# show interface fastethernet 0
FastEthernet0 is up, line protocol is up
  Hardware is epif_port, address is 000b.fcfa.339e (bia 000b.fcfa.339e)
  Description: 100 mbps full duplex q-in-q tunnel
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
     reliability 255/255, txload 18/255, rxload 200/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, 100Mb/s, 100BaseTX
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:00, output 00:00:00, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: weighted fair
  Output queue: 0/1000/64/0 (size/max total/threshold/drops)
     Conversations  0/0/256 (active/max active/max total)
     Reserved Conversations 0/0 (allocated/max allocated)
     Available Bandwidth 75000 kilobits/sec
  30 second input rate 78525000 bits/sec, 144348 packets/sec
  30 second output rate 7363000 bits/sec, 13537 packets/sec
```

```
4095063706 packets input, 3885007012 bytes
Received 0 broadcasts (0 IP multicast)
2 runts, 0 giants, 0 throttles
4 input errors, 0 CRC, 0 frame, 1 overrun, 0 ignored
0 watchdog, 0 multicast
0 input packets with dribble condition detected
1463732665 packets output, 749573412 bytes, 0 underruns
131072 output errors, 131072 collisions, 0 interface resets
0 babbles, 0 late collision, 0 deferred
0 lost carrier, 0 no carrier
0 output buffer failures, 0 output buffers swapped out
```

Enter the **show controller** command to display information about the Fast Ethernet controller chip.

Example 4-3 shows the output from the **show controller** command, which shows statistics, including information about initialization block information and raw MAC counters.

Example 4-3 show controller Command Output

```
ML_Series# show controller fastethernet 0
IF Name: FastEthernet0
Port Status UP
Send Flow Control    : Disabled
Receive Flow Control : Enabled

MAC registers
CMCR : 0x00000433 (Tx Enabled, Rx Enabled)
CMPR : 0x150B0A82 (Long Frame Enabled)
FCR  : 0x00008007

MII registers:

Control Register              (0x0): 0x100  (Auto negotation disabled)
Status Register               (0x1): 0x780D (Link status Up)
PHY Identification Register 1 (0x2): 0x40
PHY Identification Register 2 (0x3): 0x61D4
Auto Neg. Advertisement Reg   (0x4): 0x461  (Speed 10, Duplex Full)
Auto Neg. Partner Ability Reg (0x5): 0x0    (Speed 10, Duplex Half)
Auto Neg. Expansion Register  (0x6): 0x4
100Base-X Aux Control Reg    (0x10): 0x0
100Base-X Aux Status Register(0x11): 0x0
100Base-X Rcv Error Counter  (0x12): 0x0
100Base-X False Carr. Counter(0x13): 0x400
100Base-X Disconnect Counter (0x14): 0x200
Aux Control/Status Register  (0x18): 0x31
Aux Status Summary Register  (0x19): 0x5
Interrupt Register           (0x1A): 0xC000
10Base-T Aux Err & Gen Status(0x1C): 0x3021
Aux Mode Register            (0x1D): 0x0
Aux Multi-phy Register       (0x1E): 0x0

Counters :
MAC receive conters:
Bytes                     749876721
pkt64                     2394
pkts64to127               49002
pkts128to255              21291
pkts256to511              11308
pkts512to1023             40175
pkts1024to1518            24947
pkts1519to1530            54893
pkts_good_giants          11319
pkts_error_giants         0
```

```
pkts_good_runts           0
pkts_error_runts          5
pkts_ucast                26976
pkts_mcast                57281
pkts_bcast                0
align_errors              1
FCS_errors                5
Overruns                  0

MAC Transmit Counters
Bytes                     1657084026
pkts64                    23344
pkts65to127               48188
pkts128to255              12358
pkts256to511              38550
pkts512to1023             24897
pkts1024to1518            11305
pkts1519to1530            62760
pkts_ucast                17250
pkts_mcast                23108
pkts_bcast                11
pkts_fcs_err              0
pkts_giants               0
pkts_underruns            0
pkts_one_collision        0
pkts_multiple_collisions  0
pkts_excessive_collision  0
Ucode drops               2053079661
```

Enter the **show run interface** [*type number*] command to display information about the configuration of the Fast Ethernet interface. The command is useful when there are multiple interfaces and you want to look at the configuration of a specific interface.

Example 4-4 shows output from the **show run interface** [*type number*] command, which includes information about the IP or lack of IP address and the state of the interface.

Example 4-4 show run interface Command Output

```
daytona# show run interface fastethernet 1
Building configuration...

Current configuration : 222 bytes
!
interface FastEthernet1
 no ip address
 duplex full
 speed 10
 mode dot1q-tunnel
 l2protocol-tunnel cdp
 l2protocol-tunnel stp
 l2protocol-tunnel vtp
 no cdp enable
 bridge-group 2
 bridge-group 2 spanning-disabled
end
```

CHAPTER 5

Configuring POS on the ML-Series Card

This chapter describes advanced packet-over-SONET (POS) interface configuration for the ML-Series card. Basic POS interface configuration is included in Chapter 4, "Configuring Interfaces on the ML-Series Card." For more information about the Cisco IOS commands used in this chapter, refer to the *Cisco IOS Command Reference* publication.

This chapter contains the following major sections:

Understanding POS on the ML-Series Card

Ethernet frames and IP data packets need to be framed and encapsulated into SONET frames for transport across the SONET network. This framing and encapsulation process is known as POS and is carried out by the ML-Series card.

The ML-Series card treats all the standard Ethernet ports on the front of the card and the two POS ports as switch ports. Under Cisco IOS, the POS port is an interface similar to the other Ethernet interfaces on the ML-Series card. Many standard Cisco IOS features, such as IEEE 802.1 Q VLAN configuration, are configured on the POS interface in the same manner as on a standard Ethernet interface. Other features and configurations are done strictly on the POS interface. The configuration of features limited to POS ports is shown in this chapter.

Available Circuit Sizes and Combinations

Each POS port terminates an independent contiguous SONET concatenation (CCAT) or virtual SONET concatenation (VCAT). The SONET circuit is created for these ports through Cisco Transport Controller (CTC) or Transaction Language One (TL1) in the same manner as a SONET circuit is created for a non-Ethernet line card. Table 5-1shows the circuit sizes available for the ML-Series card on the ONS 15310-CL and ONS 15310-MA, and the circuit sizes required for Ethernet wire speeds.

Table 5-1 ML-Series Card Supported Circuit Sizes and Sizes Required for Ethernet Wire Speeds

Ethernet Wire Speed	CCAT High Order	VCAT High Order
10 Mbps	STS-1	STS-1-1v
100 Mbps	—	STS-1-2v[1]

1. STS-1-2v provides a total transport capacity of 98 Mbps

Caution The maximum tolerable VCAT differential delay for the ML-100T-8 is 48 milliseconds. The VCAT differential delay is the relative arrival time measurement between members of a virtual concatenation group (VCG).

Note The initial state of the ONS 15310-CL and ONS 15310-MA ML-Series card POS port is inactive. A POS interface command of **no shutdown** is required to carry traffic on the SONET circuit.

Note ML-Series card POS interfaces normally send an alarm for signal label mismatch failure in the ONS 15454 STS path overhead (PDI-P) to the far end when the POS link goes down or when RPR wraps. ML-Series card POS interfaces do not send PDI-P to the far-end when PDI-P is detected, when a remote defection indication alarm (RDI-P) is being sent to the far end, or when the only defects detected are generic framing procedure (GFP)-loss of frame delineation (LFD), GFP client signal fail (CSF), virtual concatenation (VCAT)-loss of multiframe (LOM), or VCAT-loss of sequence (SQM).

LCAS Support

The ML-100T-8 card and the CE-100T-8 card (both the ONS 15310-CL/ONS 15310-MA version and the ONS 15454 SONET/SDH version) have hardware-based support for the ITU-T G.7042 standard link capacity adjustment scheme (LCAS). This allows the user to dynamically resize a high-order or low-order VCAT circuit through CTC or TL1 without affecting other members of the VCG (errorless). ML-100T-8 LCAS support is high order only and is limited to a two-member VCG.

The ONS 15454 SONET/SDH ML-Series card has a software-based LCAS (SW-LCAS) scheme. This scheme is also supported by both the ML-100T-8 card and both versions of the CE-100T-8, but only for circuits terminating on an ONS 15454 SONET ML-Series card.

J1 Path Trace, and SONET Alarms

The ML-100T-8 card also reports SONET alarms and transmits and monitors the J1 path trace byte in the same manner as OC-N cards. Support for path termination functions includes:

- H1 and H2 concatenation indication
- Bit interleaved parity 3 (BIP-3) generation
- G1 path status indication
- C2 path signal label read/write

- Path-level alarms and conditions, including loss of pointer (LOP), unequipped (UNEQ-P), payload mismatch (PLM-P), alarm indication signal (AIS) detection, and remote defect indication (RDI)
- J1 path trace for high-order paths

Framing Mode, Encapsulation, Scrambling, MTU and CRC Support

The ML-Series card on the ONS 15310-CL and ONS 15310-MA supports high-level data link control (HDLC) framing and frame-mapped generic framing procedure (GFP-F) framing. Supported encapsulation and cyclic redundancy check (CRC) sizes for the framing types are detailed in Table 5-2.

Table 5-2 ML-Series Card Encapsulation, Framing, and CRC Sizes

	GFP-F Framing	HDLC Framing
Encapsulations	LEX (default)[1] Cisco HDLC PPP/BCP	LEX (default)
CRC Sizes	32-bit (default)	32-bit (default) None (FCS disabled)

1. RPR requires LEX encapsulation in either framing mode.

LEX is the common term for Cisco-EoS-LEX, which is a proprietary Cisco Ethernet-over-SONET encapsulation. This encapsulation is available on most ONS Ethernet cards. When the ML-Series card is configured for GFP-F framing, the LEX encapsulation is in accordance with ITU-T G.7041 as standard mapped Ethernet over GFP. Under GFP-F framing, the Cisco IOS CLI also uses this lex keyword to represent standard mapped Ethernet over GFP-F.

LEX encapsulation is the required and default encapsulation for RPR on the ML-Series card. The maximum transmission unit (MTU) size is not configurable and is set at a 1500-byte maximum (standard Ethernet MTU). In addition, the ML-Series card supports baby giant frames in which the standard Ethernet frame is augmented by IEEE 802.1 Q tags or Multiprotocol Label Switching (MPLS) tags. It does not support full Jumbo frames.

The ML-Series card supports GFP null mode. GFP-F client-management frames (CMFs) are counted and discarded.

The ML-100T-8 card is interoperable with the ONS 15310-CL and ONS 15310-MA CE-100T-8 card and several other ONS Ethernet cards. For specific details on the ONS 15310-CL and ONS 15310-MA CE-100T-8 card's encapsulation, framing, and CRC, see Chapter 17, "CE-100T-8 Ethernet Operation." For specific details on interoperability with other ONS system Ethernet cards, including framing mode, encapsulation, and CRC, refer to the "POS on ONS Ethernet Cards" chapter of the *Cisco ONS 15454 and Cisco ONS 15454 SDH Ethernet Card Software Feature and Configuration Guide*.

Configuring the POS Interface

The user can configure framing mode, encapsulation, and Cisco IOS SONET alarm reporting parameters through Cisco IOS.

Scrambling on the ONS 15310-CL and ONS 15310-MA ML-Series card is on by default and is not configurable. The C2 byte is not configurable. CRC-under-HDLC framing is restricted to 32-bit and is not configurable. CRC-under-GFP-F is restricted to 32-bit, but can be enabled (default) and disabled.

Note ML-Series card POS interfaces normally send PDI-P to the far end when the POS link goes down or RPR wraps. ML-Series card POS interfaces do not send PDI-P to the far end when PDI-P is detected, when RDI-P is being sent to the far end, or when the only defects detected are GFP LFD, GFP CSF, VCAT LOM, or VCAT SQM.

Configuring POS Interface Framing Mode

You can configure framing mode on an ML-100T-8 card through Cisco IOS. You cannot configure framing mode through CTC on the ML-100T-8 card.

Framing mode can be changed on a port by port basis. The user does not need to delete the existing circuits or reboot the ML-100T-8 card. On the ONS 15454 or ONS 15454 SDH ML-Series cards, the circuits must be deleted and the card must reboot for the framing mode to change.

To configure framing mode for the ML-Series card, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface pos number`	Activates interface configuration mode to configure the POS interface.
Step 2	`Router(config-if)# shutdown`	Manually shuts down the interface. Encapsulation and framing mode changes on POS ports are allowed only when the interface is shut down (ADMIN_DOWN).

	Command	Purpose
Step 3	`Router(config-if)# [no] pos mode gfp [fcs-disabled]`	Sets the framing mode employed by the ONS Ethernet card for framing and encapsulating data packets onto the SONET transport layer. Valid framing modes are: • **HDLC**—A common mechanism employed in framing data packets for SONET. HDLC is not a keyword choice in the command. The no form of the command sets the framing mode to Cisco HDLC. • **GFP** (default)—The ML-Series card supports the frame mapped version of generic framing procedure (GFP-F). GFP-F with a 32-bit CRC, also referred to as frame check sequence (FCS), is enabled by default. The optional FCS-disabled keyword disables the GFP-F 32-bit FCS. The FCS-disabled keyword is not available when setting the framing mode to Cisco HDLC. **Note** CRC-under-HDLC framing is restricted to a 32-bit size and cannot be disabled. **Note** The GFP-F FCS is compliant with ITU-T G.7041/Y.1303
Step 4	`Router(config-if)# no shutdown`	Restarts the shutdown interface.
Step 5	`Router(config)# end`	Returns to privileged EXEC mode.
Step 6	`Router# copy running-config startup-config`	(Optional) Saves configuration changes to NVRAM.

Configuring POS Interface Encapsulation Type Under GFP-F Framing

To configure the encapsulation type for a ML-Series card, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface pos` *number*	Activates interface configuration mode to configure the POS interface.
Step 2	`Router(config-if)# shutdown`	Manually shuts down the interface. Encapsulation and framing mode changes on POS ports are allowed only when the interface is shut down (ADMIN_DOWN).

	Command	Purpose
Step 3	`Router(config-if)# encapsulation type`	Sets the encapsulation type. Valid values are: • **hdlc**—Cisco HDLC • **lex**—(default) LAN extension (Cisco-EoS-LEX), special encapsulation for use with Cisco ONS Ethernet line cards • **ppp**—Point-to-Point Protocol **Note** Under HDLC framing, the ONS 15310-CL and ONS 15310-MA ML-Series card is restricted to LEX encapsulation.
Step 4	`Router(config-if)# no shutdown`	Restarts the shutdown interface.
Step 5	`Router(config)# end`	Returns to privileged EXEC mode.
Step 6	`Router# copy running-config startup-config`	(Optional) Saves configuration changes to NVRAM.

SONET Alarms

The ML-Series cards report SONET alarms under Cisco IOS, CTC, and TL1. A number of path alarms are reported in the Cisco IOS console. Configuring Cisco IOS console alarm reporting has no effect on CTC and TL1 alarm reporting. The "Configuring SONET Alarms" section on page 5-6 procedure specifies the alarms reported to the Cisco IOS console.

CTC and TL1 have sophisticated SONET alarm reporting capabilities. The ML-Series card reports Telcordia GR-253 SONET alarms on the Alarms tab of CTC, and in TL1-like other ONS system line cards. For more information about alarms and alarm definitions, refer to the "Alarm Troubleshooting" chapter of the *Cisco ONS 15454 Troubleshooting Guide*.

Configuring SONET Alarms

All SONET alarms are logged on the Cisco IOS CLI by default. But to provision or disable the reporting of SONET alarms on the Cisco IOS CLI, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface pos` *number*	Enters interface configuration mode and specifies the POS interface to configure.
Step 2	`Router(config-if)# pos report {all | encap | pais | plop | ppdi | pplm | prdi | ptim | puneq | sd-ber-b3 | sf-ber-b3}`	Permits console logging of selected SONET alarms. Use the **no** form of the command to disable reporting of a specific alarm. The alarms are as follows: • **all—All alarms/signals** • encap—Path encapsulation mismatch • **pais**—Path alarm indication signal • **plop**—Path loss of pointer • **ppdi**—Path payload defect indication • **pplm**—Payload label, C2 mismatch • **prdi**—Path remote defect indication • **ptim**—Path trace identifier mismatch • **puneq**—Path label equivalent to zero • **sd-ber-b3**—PBIP BER in excess of SD threshold • **sf-ber-b3**—PBIP BER in excess of SF threshold
Step 3	`Router(config-if)# end`	Returns to the privileged EXEC mode.
Step 4	`Router# copy running-config startup-config`	(Optional) Saves configuration changes to NVRAM.

To determine which alarms are reported on the POS interface and to display the bit error rate (BER) thresholds, use the **show controllers pos** command, as described in the "Monitoring and Verifying POS" section on page 5-8.

Configuring SONET Delay Triggers

You can set path alarms listed as triggers to bring down the line protocol of the POS interface. When you configure the path alarms as triggers, you can also specify a delay for the triggers using the **pos trigger delay** command. You can set the delay from 200 to 2000 ms. If you do not specify a time interval, the default delay is set to 200 ms.

To configure path alarms as triggers and specify a delay, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface pos` *number*	Enters interface configuration mode and specifies the POS interface to configure.
Step 2	`Router(config-if)# pos trigger defect {all | ber_sf_b3 | encap | pais | plop | ppdi | pplm | prdi | ptim | puneq}`	Configures certain path defects as triggers to bring down the POS interface. The configurable triggers are as follows: • **all**—All link down alarm failures • **ber_sd_b3**—PBIP BER in excess of SD threshold failure • **ber_sf_b3**—PBIP BER in excess of SD threshold failure (default) • **encap**—Path Signal Label Encapsulation Mismatch failure (default) • **pais**—Path Alarm Indication Signal failure (default) • **plop**—Path Loss of Pointer failure (default) • **ppdi**—Path Payload Defect Indication failure (default) • **pplm**—Payload label mismatch path (default) • **prdi**—Path Remote Defect Indication failure (default) • **ptim**—Path Trace Indicator Mismatch failure (default) • **puneq**—Path Label Equivalent to Zero failure (default)
Step 3	`Router(config-if)# pos trigger delay` *millisecond*	Sets waiting period before the line protocol of the interface goes down. Delay can be set from 200 to 2000 ms. If no time intervals are specified, the default delay is set to 200 ms.
Step 4	`Router(config-if)# end`	Returns to the privileged EXEC mode.
Step 5	`Router# copy running-config startup-config`	(Optional) Saves configuration changes to NVRAM.

Monitoring and Verifying POS

Showing the outputs framing mode and concatenation information with the **show controller pos** [**0** | **1**] command (Example 5-1).

Example 5-1 Showing Framing Mode and Concatenation Information with the show controller pos [0 | 1] Command

```
ML_Series# show controller pos0
Interface POS0
Hardware is Packet Over SONET
Framing Mode: HDLC
Concatenation: CCAT
***************  GFP  ***************
Active Alarms : None
Active Alarms : None
  LDF      = 0          CSF        = 0
```

```
CCAT/VCAT info not available yet!

56517448726 total input packets,  4059987309747 post-encap bytes
0 input short packets,  ?? pre-encap bytes
283  input CRCerror packets , 0  input drop packets
564  rx HDLC addr mismatchs , 564  rx HDLC ctrl mismatchs
564  rx HDLC sapi mismatchs , 564  rx HDLC ctrl mismatchs
0  rx HDLC destuff errors , 564  rx HDLC invalid frames
0 input abort packets
5049814101 input packets dropped by ucode
0 input packets congestion drops
56733042489 input good packets (POS MAC rx)
4073785395967 input good octets (POS MAC rx)


56701415757 total output packets,  4059987309747 post-encap bytes

Carrier delay is 200 msec
```

Showing scrambling with the **show interface pos** [**0 | 1**] command (Example 5-2).

Example 5-2 Showing Scrambling with the show interface pos [0 | 1] Command

```
ML_Series# show interface pos 0
POS0 is up, line protocol is down
    Hardware is Packet Over SONET, address is 000b.fcfa.33b0 (bia 000b.fcfa.33b0)
    MTU 1500 bytes, BW 48384 Kbit, DLY 100 usec,
    reliability 255/255, txload 1/255, rxload 1/255
    Encapsulation: Cisco-EoS-LEX, loopback not set
    Keepalive set (10 sec)
    Scramble enabled
    ARP type: ARPA, ARP Timeout 04:00:00
    Last input 22:46:51, output never, output hang never
    Last clearing of "show interface" counters 1w5d
    Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
    Queueing strategy: fifo
    Output queue: 0/40 (size/max)
    5 minute input rate 0 bits/sec, 0 packets/sec
    5 minute output rate 0 bits/sec, 0 packets/sec
    777 packets input, 298426 bytes
    Received 0 broadcasts (0 IP multicast)
        0 runts, 0 giants, 0 throttles
        0 parity
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
    0 input packets with dribble condition detected
    769 packets output, 296834 bytes, 0 underruns
    0 output errors, 0 applique, 1 interface resets
    0 babbles, 0 late collision, 0 deferred
    0 lost carrier, 0 no carrier
    0 output buffer failures, 0 output buffers swapped out
    0 carrier transitions
```

CHAPTER 6

Configuring STP and RSTP on the ML-Series Card

This chapter describes the IEEE 802.1D Spanning Tree Protocol (STP) and the ML-Series implementation of the IEEE 802.1W Rapid Spanning Tree Protocol (RSTP). It also explains how to configure STP and RSTP on the ML-Series card.

This chapter consists of these sections:

STP Features

These sections describe how the spanning-tree features work:

STP Overview

STP is a Layer 2 link management protocol that provides path redundancy while preventing loops in the network. For a Layer 2 Ethernet network to function properly, only one active path can exist between any two stations. Spanning-tree operation is transparent to end stations, which cannot detect whether they are connected to a single LAN segment or a switched LAN of multiple segments.

When you create fault-tolerant internetworks, you must have a loop-free path between all nodes in a network. The spanning-tree algorithm calculates the best loop-free path throughout a switched Layer 2 network. Switches send and receive spanning-tree frames, called bridge protocol data units (BPDUs), at regular intervals. The switches do not forward these frames, but use the frames to construct a loop-free path.

Multiple active paths among end stations cause loops in the network. If a loop exists in the network, end stations might receive duplicate messages. Switches might also learn end-station MAC addresses on multiple Layer 2 interfaces. These conditions result in an unstable network.

Spanning tree defines a tree with a root switch and a loop-free path from the root to all switches in the Layer 2 network. Spanning tree forces redundant data paths into a standby (blocked) state. If a network segment in the spanning tree fails and a redundant path exists, the spanning-tree algorithm recalculates the spanning-tree topology and activates the standby path.

When two interfaces on a switch are part of a loop, the spanning-tree port priority and path cost settings determine which interface is put in the forwarding state and which is put in the blocking state. The port priority value represents the location of an interface in the network topology and how well it is located to pass traffic. The path cost value represents media speed.

Supported STP Instances

The ML-Series card supports the per-VLAN spanning tree (PVST+) and a maximum of 255 spanning-tree instances.

Caution At more than 100 STP instances the STP instances may flap and may result in MAC entries flushed, and MAC entries learned again and again. This will cause flooding in the network. So it is recommended to keep the STP instances to be less than 100, to keep system from being unstable.

Bridge Protocol Data Units

The stable, active, spanning-tree topology of a switched network is determined by these elements:

- Unique bridge ID (switch priority and MAC address) associated with each VLAN on each switch
- Spanning-tree path cost to the root switch
- Port identifier (port priority and MAC address) associated with each Layer 2 interface

When the switches in a network are powered up, each functions as the root switch. Each switch sends a configuration BPDU through all of its ports. The BPDUs communicate and compute the spanning-tree topology. Each configuration BPDU contains this information:

- Unique bridge ID of the switch that the sending switch identifies as the root switch
- Spanning-tree path cost to the root
- Bridge ID of the sending switch

- Message age
- Identifier of the sending interface
- Values for the hello, forward delay, and max-age protocol timers

When a switch receives a configuration BPDU that contains superior information (lower bridge ID, lower path cost, etc.), it stores the information for that port. If this BPDU is received on the root port of the switch, the switch also forwards it with an updated message to all attached LANs for which it is the designated switch.

If a switch receives a configuration BPDU that contains inferior information to that currently stored for that port, it discards the BPDU. If the switch is a designated switch for the LAN from which the inferior BPDU was received, it sends that LAN a BPDU containing the up-to-date information stored for that port. In this way, inferior information is discarded, and superior information is propagated on the network.

A BPDU exchange results in these actions:

- One switch in the network is elected as the root switch.
- A root port is selected for each switch (except the root switch). This port provides the best path (lowest cost) when the switch forwards packets to the root switch.
- The shortest distance to the root switch is calculated for each switch based on the path cost.
- A designated switch for each LAN segment is selected. The designated switch incurs the lowest path cost when forwarding packets from that LAN to the root switch. The port through which the designated switch is attached to the LAN is called the designated port.
- Interfaces included in the spanning-tree instance are selected. Root ports and designated ports are put in the forwarding state.
- All interfaces not included in the spanning tree are blocked.

Election of the Root Switch

All switches in the Layer 2 network participating in the spanning tree gather information about other switches in the network through an exchange of BPDU data messages. This exchange of messages results in these actions:

- Election of a unique root switch for each spanning-tree instance
- Election of a designated switch for every switched LAN segment
- Removal of loops in the switched network by blocking Layer 2 interfaces connected to redundant links

For each VLAN, the switch with the highest switch priority (the lowest numerical priority value) is elected as the root switch. If all switches are configured with the default priority (32768), the switch with the lowest MAC address in the VLAN becomes the root switch. The switch priority value occupies the most significant bits of the bridge ID.

When you change the switch priority value, you change the probability that the switch will be elected as the root switch. Configuring a higher value decreases the probability; a lower value increases the probability.

The root switch is the logical center of the spanning-tree topology in a switched network. All paths that are not needed to reach the root switch from anywhere in the switched network are placed in the spanning-tree blocking mode.

BPDUs contain information about the sending switch and its ports, including switch and MAC addresses, switch priority, port priority, and path cost. Spanning tree uses this information to elect the root switch and root port for the switched network and the root port and designated port for each switched segment.

Bridge ID, Switch Priority, and Extended System ID

The IEEE 802.1D standard requires that each switch has an unique bridge identifier (bridge ID), which determines the selection of the root switch. Because each VLAN is considered as a different logical bridge with PVST+, the same switch must have as many different bridge IDs as VLANs configured on it. Each VLAN on the switch has a unique 8-byte bridge ID; the two most-significant bytes are used for the switch priority, and the remaining six bytes are derived from the switch MAC address.

The ML-Series card supports the IEEE 802.1T spanning-tree extensions, and some of the bits previously used for the switch priority are now used as the bridge ID. The result is that fewer MAC addresses are reserved for the switch, and a larger range of VLAN IDs can be supported, all while maintaining the uniqueness of the bridge ID. As shown in Table 6-1, the two bytes previously used for the switch priority are reallocated into a 4-bit priority value and a 12-bit extended system ID value equal to the bridge ID. In earlier releases, the switch priority is a 16-bit value.

Table 6-1 Switch Priority Value and Extended System ID

Switch Priority Value				Extended System ID (Set Equal to the Bridge ID)											
Bit 16	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1
32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1

Spanning tree uses the extended system ID, the switch priority, and the allocated spanning-tree MAC address to make the bridge ID unique for each VLAN.

Spanning-Tree Timers

Table 6-2 describes the timers that affect the entire spanning-tree performance.

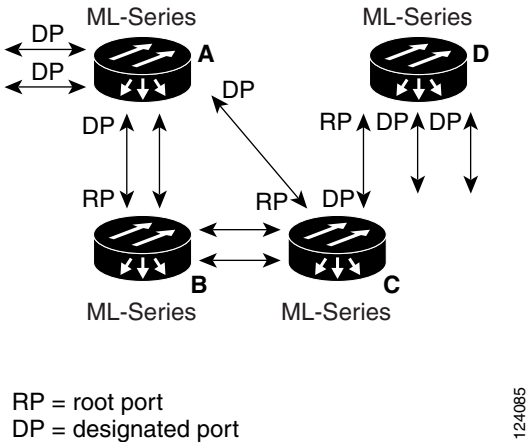
Table 6-2 Spanning-Tree Timers

Variable	Description
Hello timer	When this timer expires, the interface sends out a Hello message to the neighboring nodes.
Forward-delay timer	Determines how long each of the listening and learning states last before the interface begins forwarding.
Maximum-age timer	Determines the amount of time the switch stores protocol information received on an interface.

Creating the Spanning-Tree Topology

In Figure 6-1, Switch A is elected as the root switch because the switch priority of all the switches is set to the default (32768) and Switch A has the lowest MAC address. However, because of traffic patterns, number of forwarding interfaces, or link types, Switch A might not be the ideal root switch. By increasing the priority (lowering the numerical value) of the ideal switch so that it becomes the root switch, you force a spanning-tree recalculation to form a new topology with the ideal switch as the root.

Figure 6-1 Spanning-Tree Topology



When the spanning-tree topology is calculated based on default parameters, the path between source and destination end stations in a switched network might not be ideal. For instance, connecting higher-speed links to an interface that has a higher number than the root port can cause a root-port change. The goal is to make the fastest link the root port.

Spanning-Tree Interface States

Propagation delays can occur when protocol information passes through a switched LAN. As a result, topology changes can take place at different times and at different places in a switched network. When an interface transitions directly from nonparticipation in the spanning-tree topology to the forwarding state, it can create temporary data loops. Interfaces must wait for new topology information to propagate through the switched LAN before starting to forward frames. They must allow the frame lifetime to expire for forwarded frames that have used the old topology.

Each Layer 2 interface on a switch using spanning tree exists in one of these states:

- Blocking—The interface does not participate in frame forwarding.
- Listening—The first transitional state after the blocking state when the spanning tree determines that the interface should participate in frame forwarding.
- Learning—The interface prepares to participate in frame forwarding.
- Forwarding—The interface forwards frames.
- Disabled—The interface is not participating in spanning tree because of a shutdown port, no link on the port, or no spanning-tree instance running on the port.

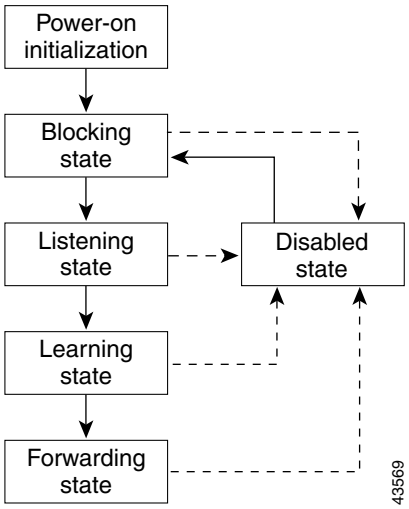
An interface moves through these states:

1. From initialization to blocking

2. From blocking to listening or to disabled
3. From listening to learning or to disabled
4. From learning to forwarding or to disabled
5. From forwarding to disabled

Figure 6-2 illustrates how an interface moves through the states.

Figure 6-2 Spanning-Tree Interface States



When you power up the switch, STP is enabled by default, and every interface in the switch, VLAN, or network goes through the blocking state and the transitory states of listening and learning. Spanning tree stabilizes each interface at the forwarding or blocking state.

When the spanning-tree algorithm places a Layer 2 interface in the forwarding state, this process occurs:

1. The interface is in the listening state while spanning tree waits for protocol information to transition the interface to the blocking state.
2. While spanning tree waits for the forward-delay timer to expire, it moves the interface to the learning state and resets the forward-delay timer.
3. In the learning state, the interface continues to block frame forwarding as the switch learns end-station location information for the forwarding database.
4. When the forward-delay timer expires, spanning tree moves the interface to the forwarding state, where both learning and frame forwarding are enabled.

Blocking State

A Layer 2 interface in the blocking state does not participate in frame forwarding. After initialization, a BPDU is sent to each interface in the switch. A switch initially functions as the root until it exchanges BPDUs with other switches. This exchange establishes which switch in the network is the root or root switch. If there is only one switch in the network, no exchange occurs, the forward-delay timer expires, and the interfaces move to the listening state. An interface always enters the blocking state after switch initialization.

An interface in the blocking state performs as follows:

- Discards frames received on the port

- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Receives BPDUs

Listening State

The listening state is the first state a Layer 2 interface enters after the blocking state. The interface enters this state when the spanning tree determines that the interface should participate in frame forwarding.

An interface in the listening state performs as follows:

- Discards frames received on the port
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Receives BPDUs

Learning State

A Layer 2 interface in the learning state prepares to participate in frame forwarding. The interface enters the learning state from the listening state.

An interface in the learning state performs as follows:

- Discards frames received on the port
- Discards frames switched from another interface for forwarding
- Learns addresses
- Receives BPDUs

Forwarding State

A Layer 2 interface in the forwarding state forwards frames. The interface enters the forwarding state from the learning state.

An interface in the forwarding state performs as follows:

- Receives and forwards frames received on the port
- Forwards frames switched from another port
- Learns addresses
- Receives BPDUs

Disabled State

A Layer 2 interface in the disabled state does not participate in frame forwarding or in the spanning tree. An interface in the disabled state is nonoperational.

A disabled interface performs as follows:

- Forwards frames switched from another interface for forwarding
- Learns addresses
- Does not receive BPDUs

Spanning-Tree Address Management

IEEE 802.1D specifies 17 multicast addresses, ranging from 0x00180C2000000 to 0x0180C2000010, to be used by different bridge protocols. These addresses are static addresses that cannot be removed.

The ML-Series card switches supported BPDUs (0x0180C2000000 and 01000CCCCCCD) when they are being tunneled via the protocol tunneling feature.

STP and IEEE 802.1Q Trunks

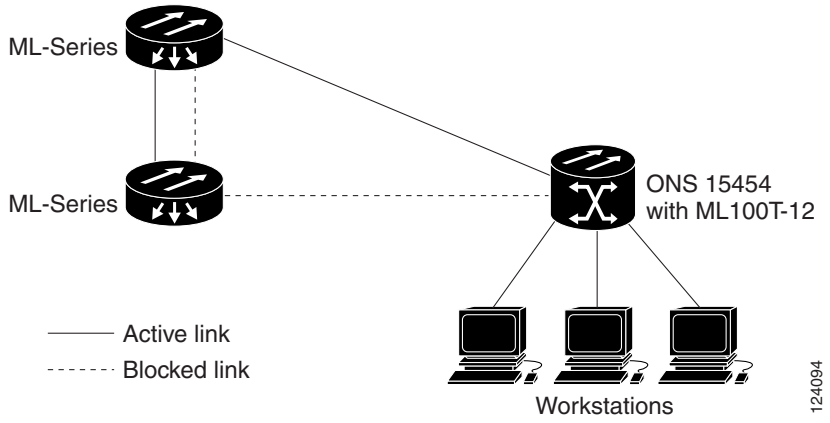
When you connect a Cisco switch to a non-Cisco device through an IEEE 802.1Q trunk, the Cisco switch uses PVST+ to provide spanning-tree interoperability. PVST+ is automatically enabled on IEEE 802.1Q trunks after users assign a protocol to a bridge group. The external spanning-tree behavior on access ports and Inter-Switch Link (ISL) trunk ports is not affected by PVST+.

For more information on IEEE 802.1Q trunks, see Chapter 7, "Configuring VLANs on the ML-Series Card."

Spanning Tree and Redundant Connectivity

You can create a redundant backbone with spanning tree by connecting two switch interfaces to another device or to two different devices. Spanning tree automatically disables one interface but enables it if the other one fails, as shown in Figure 6-3. If one link is high speed and the other is low speed, the low-speed link is always disabled. If the speeds are the same, the port priority and port ID are added together, and spanning tree disables the link with the lowest value.

Figure 6-3 Spanning Tree and Redundant Connectivity



You can also create redundant links between switches by using EtherChannel groups. For more information, see Chapter 9, "Configuring Link Aggregation on the ML-Series Card."

Accelerated Aging to Retain Connectivity

The default for aging dynamic addresses is 5 minutes, which is the default setting of the **bridge** *bridge-group-number* **aging-time** global configuration command. However, a spanning-tree reconfiguration can cause many station locations to change. Because these stations could be unreachable for 5 minutes or more during a reconfiguration, the address-aging time is accelerated so that station addresses can be dropped from the address table and then relearned.

Because each VLAN is a separate spanning-tree instance, the switch accelerates aging on a per-VLAN basis. A spanning-tree reconfiguration on one VLAN can cause the dynamic addresses learned on that VLAN to be subject to accelerated aging. Dynamic addresses on other VLANs can be unaffected and remain subject to the aging interval entered for the switch.

RSTP Features

RSTP provides rapid convergence of the spanning tree. It improves the fault tolerance of the network because a failure in one instance (forwarding path) does not affect other instances (forwarding paths). The most common initial deployment of RSTP is in the backbone and distribution layers of a Layer 2 switched network; this deployment provides the highly available network required in a service-provider environment.

RSTP improves the operation of the spanning tree while maintaining backward compatibility with equipment that is based on the (original) IEEE 802.1D spanning tree.

RSTP takes advantage of point-to-point wiring and provides rapid convergence of the spanning tree. Reconfiguration of the spanning tree can occur in less than 2 second (in contrast to 50 seconds with the default settings in the IEEE 802.1D spanning tree), which is critical for networks carrying delay-sensitive traffic such as voice and video.

These sections describe how RSTP works:

- Supported RSTP Instances, page 6-9
- Port Roles and the Active Topology, page 6-10
- Rapid Convergence, page 6-11
- Synchronization of Port Roles, page 6-12
- Bridge Protocol Data Unit Format and Processing, page 6-13
- Topology Changes, page 6-14

Supported RSTP Instances

The ML Series supports per-VLAN rapid spanning tree (PVRST) and a maximum of 255 rapid spanning-tree instances.

Caution At more than 100 RSTP instances the RSTP instances may flap and may result in MAC entries flushed, and MAC entries learned again and again. This will cause flooding in the network.. So it is recommended to keep the RSTP instances to be less than 100, to keep system from being unstable.

Port Roles and the Active Topology

The RSTP provides rapid convergence of the spanning tree by assigning port roles and by determining the active topology. The RSTP builds upon the IEEE 802.1D STP to select the switch with the highest switch priority (lowest numerical priority value) as the root switch as described in the "Election of the Root Switch" section on page 6-3. Then the RSTP assigns one of these port roles to individual ports:

- Root port—Provides the best path (lowest cost) when the switch forwards packets to the root switch.
- Designated port—Connects to the designated switch, which incurs the lowest path cost when forwarding packets from that LAN to the root switch. The port through which the designated switch is attached to the LAN is called the designated port.
- Alternate port—Offers an alternate path toward the root switch to that provided by the current root port.
- Backup port—Acts as a backup for the path provided by a designated port toward the leaves of the spanning tree. A backup port can exist only when two ports are connected together in a loopback by a point-to-point link or when a switch has two or more connections to a shared LAN segment.
- Disabled port—Has no role within the operation of the spanning tree.

A port with the root or a designated port role is included in the active topology. A port with the alternate or backup port role is excluded from the active topology.

In a stable topology with consistent port roles throughout the network, the RSTP ensures that every root port and designated port immediately transition to the forwarding state while all alternate and backup ports are always in the discarding state (equivalent to blocking in IEEE 802.1D). The port state controls the operation of the forwarding and learning processes. Table 6-3 provides a comparison of IEEE 802.1D and RSTP port states.

Table 6-3 Port State Comparison

Operational Status	STP Port State	RSTP Port State	Is Port Included in the Active Topology?
Enabled	Blocking	Discarding	No
Enabled	Listening	Discarding	No
Enabled	Learning	Learning	Yes
Enabled	Forwarding	Forwarding	Yes
Disabled	Disabled	Discarding	No

Caution STP edge ports are bridge ports that do not need STP enabled, where loop protection is not needed out of that port or an STP neighbor does not exist out of that port. For RSTP, it is important to disable STP on edge ports, which are typically front-side Ethernet ports, using the command **bridge** *bridge-group-number* **spanning-disabled** on the appropriate interface. If RSTP is not disabled on edge ports, convergence times will be excessive for packets traversing those ports.

Note To be consistent with Cisco STP implementations, Table 6-3 describes the port state as blocking instead of discarding. Designated ports start in the listening state.

Rapid Convergence

The RSTP provides for rapid recovery of connectivity following the failure of switch, a switch port, or a LAN. It provides rapid convergence for new root ports, and ports connected through point-to-point links as follows:

- Root ports—If the RSTP selects a new root port, it blocks the old root port and immediately transitions the new root port to the forwarding state.
- Point-to-point links—If you connect a port to another port through a point-to-point link and the local port becomes a designated port, it negotiates a rapid transition with the other port by using the proposal-agreement handshake to ensure a loop-free topology.

As shown in Figure 6-4, Switch A is connected to Switch B through a point-to-point link, and all of the ports are in the blocking state. Assume that the priority of Switch A is a smaller numerical value than the priority of Switch B. Switch A sends a proposal message (a configuration BPDU with the proposal flag set) to Switch B, proposing itself as the designated switch.

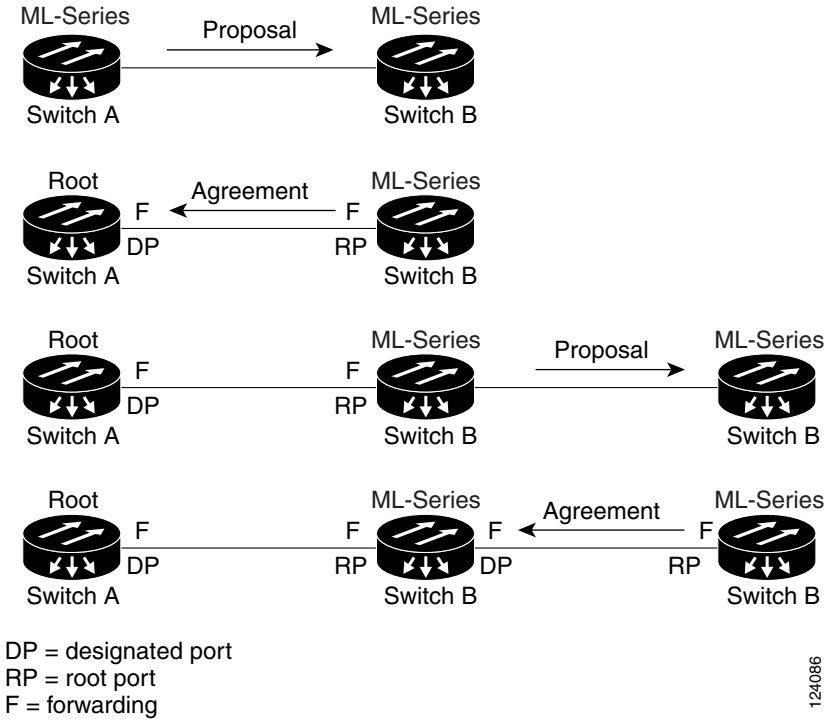
After receiving the proposal message, Switch B selects as its new root port the port from which the proposal message was received, forces all non-edge ports to the blocking state, and sends an agreement message (a BPDU with the agreement flag set) through its new root port.

After receiving an agreement message from Switch B, Switch A also immediately transitions its designated port to the forwarding state. No loops in the network are formed because Switch B blocked all of its non-edge ports and because there is a point-to-point link between Switches A and B.

When Switch C is connected to Switch B, a similar set of handshaking messages are exchanged. Switch C selects the port connected to Switch B as its root port, and both ends immediately transition to the forwarding state. With each iteration of this handshaking process, one more switch joins the active topology. As the network converges, this proposal-agreement handshaking progresses from the root toward the leaves of the spanning tree.

The switch determines the link type from the port duplex mode: a full-duplex port is considered to have a point-to-point connection; a half-duplex port is considered to have a shared connection.

Figure 6-4 Proposal and Agreement Handshaking for Rapid Convergence



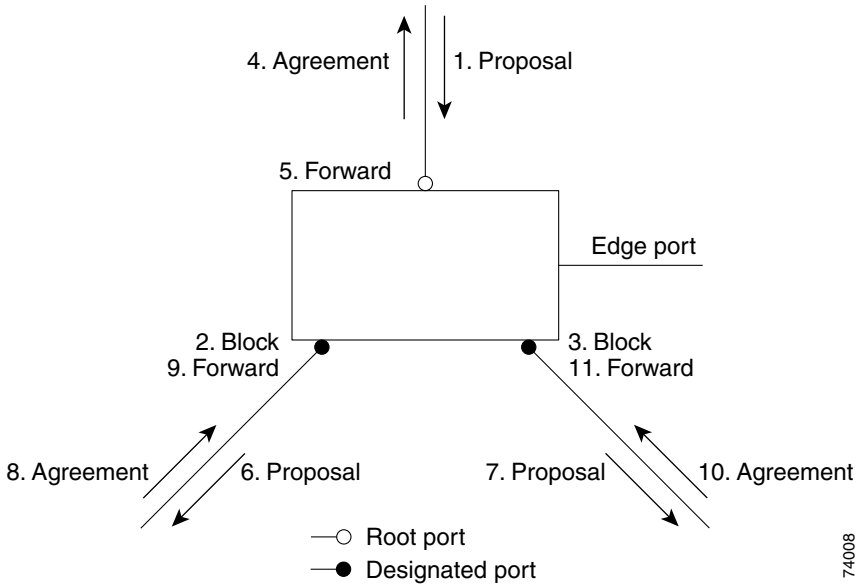
Synchronization of Port Roles

When the switch receives a proposal message on one of its ports and that port is selected as the new root port, the RSTP forces all other ports to synchronize with the new root information. The switch is synchronized with superior root information received on the root port if all other ports are synchronized.

If a designated port is in the forwarding state, it transitions to the blocking state when the RSTP forces it to synchronize with new root information. In general, when the RSTP forces a port to synchronize with root information and the port does not satisfy any of the above conditions, its port state is set to blocking.

After ensuring all of the ports are synchronized, the switch sends an agreement message to the designated switch corresponding to its root port. When the switches connected by a point-to-point link are in agreement about their port roles, the RSTP immediately transitions the port states to forwarding. The sequence of events is shown in Figure 6-5.

Figure 6-5 Sequence of Events During Rapid Convergence



Bridge Protocol Data Unit Format and Processing

The RSTP BPDU format is the same as the IEEE 802.1D BPDU format except that the protocol version is set to 2. A new Length field is set to zero, which means that no version 1 protocol information is present. Table 6-4 shows the RSTP flag fields.

Table 6-4 RSTP BPDU Flags

Bit	Function
0	Topology change (TC)
1	Proposal
2–3:	Port role:
00	Unknown
01	Alternate port
10	Root port
11	Designated port
4	Learning
5	Forwarding
6	Agreement
7	Topology change acknowledgement

The sending switch sets the proposal flag in the RSTP BPDU to propose itself as the designated switch on that LAN. The port role in the proposal message is always set to the designated port.

The sending switch sets the agreement flag in the RSTP BPDU to accept the previous proposal. The port role in the agreement message is always set to the root port.

The RSTP does not have a separate topology change notification (TCN) BPDU. It uses the topology change (TC) flag to show the topology changes. However, for interoperability with IEEE 802.1D switches, the RSTP switch processes and generates TCN BPDUs.

The learning and forwarding flags are set according to the state of the sending port.

Processing Superior BPDU Information

If a port receives superior root information (lower bridge ID, lower path cost, etc.) than currently stored for the port, the RSTP triggers a reconfiguration. If the port is proposed and is selected as the new root port, RSTP forces all the other ports to synchronize.

If the BPDU received is an RSTP BPDU with the proposal flag set, the switch sends an agreement message after all of the other ports are synchronized. If the BPDU is an IEEE 802.1D BPDU, the switch does not set the proposal flag and starts the forward-delay timer for the port. The new root port requires twice the forward-delay time to transition to the forwarding state.

If the superior information received on the port causes the port to become a backup or alternate port, RSTP sets the port to the blocking state but does not send the agreement message. The designated port continues sending BPDUs with the proposal flag set until the forward-delay timer expires, at which time the port transitions to the forwarding state.

Processing Inferior BPDU Information

If a designated port receives an inferior BPDU (higher bridge ID, higher path cost, etc.) than currently stored for the port with a designated port role, it immediately replies with its own information.

Topology Changes

This section describes the differences between the RSTP and the IEEE 802.1D in handling spanning-tree topology changes.

- Detection—Unlike IEEE 802.1D, in which any transition between the blocking and the forwarding state causes a topology change, only transitions from the blocking to the forwarding state cause a topology change with RSTP. (Only an increase in connectivity is considered a topology change.) State changes on an edge port do not cause a topology change. When an RSTP switch detects a topology change, it flushes the learned information on all of its non-edge ports.
- Notification—Unlike IEEE 802.1D, which uses TCN BPDUs, the RSTP does not use them. However, for IEEE 802.1D interoperability, an RSTP switch processes and generates TCN BPDUs.
- Acknowledgement—When an RSTP switch receives a TCN message on a designated port from an IEEE 802.1D switch, it replies with an IEEE 802.1D configuration BPDU with the topology change acknowledgement bit set. However, if the timer (the same as the topology-change timer in IEEE 802.1D) is active on a root port connected to an IEEE 802.1D switch and a configuration BPDU with the topology change acknowledgement bit set is received, the timer is reset.

  This behavior is only required to support IEEE 802.1D switches. The RSTP BPDUs never have the topology change acknowledgement bit set.

- Propagation—When an RSTP switch receives a TC message from another switch through a designated or root port, it propagates the topology change to all of its non-edge, edge, designated ports, and root port (excluding the port on which it is received). The switch starts the TC-while timer for all such ports and flushes the information learned on them.
- Protocol migration—For backward compatibility with IEEE 802.1D switches, RSTP selectively sends IEEE 802.1D configuration BPDUs and TCN BPDUs on a per-port basis.

  When a port is initialized, the timer is started (which specifies the minimum time during which RSTP BPDUs are sent), and RSTP BPDUs are sent. While this timer is active, the switch processes all BPDUs received on that port and ignores the protocol type.

  If the switch receives an IEEE 802.1D BPDU after the port's migration-delay timer has expired, it assumes that it is connected to an IEEE 802.1D switch and starts using only IEEE 802.1D BPDUs. However, if the RSTP switch is using IEEE 802.1D BPDUs on a port and receives an RSTP BPDU after the timer has expired, it restarts the timer and starts using RSTP BPDUs on that port.

Interoperability with IEEE 802.1D STP

A switch running RSTP supports a built-in protocol migration mechanism that enables it to interoperate with legacy IEEE 802.1D switches. If this switch receives a legacy IEEE 802.1D configuration BPDU (a BPDU with the protocol version set to 0), it sends only IEEE 802.1D BPDUs on that port.

However, the switch does not automatically revert to the RSTP mode if it no longer receives IEEE 802.1D BPDUs because it cannot determine whether the legacy switch has been removed from the link unless the legacy switch is the designated switch. Also, a switch might continue to assign a boundary role to a port when the switch to which this switch is connected has joined the region.

Configuring STP and RSTP Features

These sections describe how to configure spanning-tree features:

- Default STP and RSTP Configuration, page 6-16
- Disabling STP and RSTP, page 6-16
- Configuring the Root Switch, page 6-17
- Configuring the Port Priority, page 6-17
- Configuring the Path Cost, page 6-18
- Configuring the Switch Priority of a Bridge Group, page 6-18
- Configuring the Hello Time, page 6-19
- Configuring the Forwarding-Delay Time for a Bridge Group, page 6-20
- Configuring the Maximum-Aging Time for a Bridge Group, page 6-20

Default STP and RSTP Configuration

Table 6-5 shows the default STP and RSTP configuration.

Table 6-5 Default STP and RSTP Configuration

Feature	Default Setting
Enable state	Up to 255 spanning-tree instances can be enabled.
Switch priority	32768 + Bridge ID
Spanning-tree port priority (configurable on a per-interface basis—used on interfaces configured as Layer 2 access ports)	128
Spanning-tree port cost (configurable on a per-interface basis)	100 Mbps: 19 10 Mbps: 100 STS-1: 37
Hello time	2 seconds
Forward-delay time	15 seconds
Maximum-aging time	20 seconds

Disabling STP and RSTP

STP is enabled by default on the native VLAN 1 and on all newly created VLANs up to the specified spanning-tree limit of 255. Disable STP only if you are sure there are no loops in the network topology.

Caution STP edge ports are bridge ports that do not need STP enabled—where loop protection is not needed out of that port or an STP neighbor does not exist out of that port. For RSTP, it is important to disable STP on edge ports, which are typically front-side Ethernet ports, using the command **bridge** *bridge-group-number* **spanning-disabled** on the appropriate interface. If RSTP is not disabled on edge ports, convergence times will be excessive for packets traversing those ports.

Caution When STP is disabled and loops are present in the topology, excessive traffic and indefinite packet duplication can drastically reduce network performance.

Beginning in privileged EXEC mode, follow these steps to disable STP or RSTP on a per-VLAN basis:

	Command	Purpose
Step 1	`ML_Series# configure terminal`	Enters the global configuration mode.
Step 2	`ML_Series(config)# interface interface-id`	Enters the interface configuration mode.
Step 3	`ML_Series(config-if)# bridge-group bridge-group-number spanning disabled`	Disables STP or RSTP on a per-interface basis.
Step 4	`ML_Series(config-if)# end`	Returns to privileged EXEC mode.

To reenable STP, use the **no bridge-group** *bridge-group-number* **spanning disabled** interface-level configuration command.

Configuring the Root Switch

The switch maintains a separate spanning-tree instance for each active VLAN configured on it. A bridge ID, consisting of the switch priority and the switch MAC address, is associated with each instance. For each VLAN, the switch with the lowest bridge ID becomes the root switch for that VLAN.

Note If your network consists of switches that both do and do not support the extended system ID, it is unlikely that the switch with the extended system ID support will become the root switch. The extended system ID increases the switch priority value every time the bridge ID is greater than the priority of the connected switches that are running older software.

Configuring the Port Priority

If a loop occurs, spanning tree uses the port priority when selecting an interface to put into the forwarding state. You can assign higher priority values (lower numerical values) to interfaces that you want selected first, and lower priority values (higher numerical values) that you want selected last. If all interfaces have the same priority value, spanning tree puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

Beginning in privileged EXEC mode, follow these steps to configure the port priority of an interface:

	Command	Purpose
Step 1	`ML_Series# configure terminal`	Enters the global configuration mode.
Step 2	`ML_Series(config)# interface interface-id`	Enters the interface configuration mode, and specifies an interface to configure. Valid interfaces include physical interfaces and port-channel logical interfaces (**port-channel** *port-channel-number*).
Step 3	`ML_Series(config-if)# bridge-group bridge-group-number priority-value`	Configures the port priority for an interface that is an access port. For the *priority-value*, the range is 0 to 255; the default is 128 in increments of 16. The lower the number, the higher the priority.
Step 4	`ML_Series(config-if)# end`	Return to privileged EXEC mode.

To return the interface to its default setting, use the **no bridge-group id** *bridge-group-number priority-value* command.

Configuring the Path Cost

The spanning-tree path cost default value is derived from the media speed of an interface. If a loop occurs, spanning tree uses cost when selecting an interface to put in the forwarding state. You can assign lower cost values to interfaces that you want selected first and higher cost values to interfaces that you want selected last. If all interfaces have the same cost value, spanning tree puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

Beginning in privileged EXEC mode, follow these steps to configure the cost of an interface:

	Command	Purpose
Step 1	`ML_Series# configure terminal`	Enters the global configuration mode.
Step 2	`ML_Series(config)# interface` *interface-id*	Enters the interface configuration mode and specifies an interface to configure. Valid interfaces include physical interfaces and port-channel logical interfaces (**port-channel** *port-channel-number*).
Step 3	`ML_Series(config-if)# bridge-group` *bridge-group-number* `path-cost` *cost*	Configures the cost for an interface that is an access port. If a loop occurs, spanning tree uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission. For *cost*, the range is 0 to 65535; the default value is derived from the media speed of the interface.
Step 4	`ML_Series(config-if)# end`	Returns to the privileged EXEC mode.

Note The **show spanning-tree interface** *interface-id* privileged EXEC command displays information only for ports that are in a link-up operative state. Otherwise, you can use the **show running-config** privileged EXEC command to confirm the configuration.

To return the interface to its default setting, use the **no bridge-group** *bridge-group-number* **path-cost** *cost* command.

Configuring the Switch Priority of a Bridge Group

You can configure the switch priority and make it more likely that the switch will be chosen as the root switch.

Beginning in privileged EXEC mode, follow these steps to configure the switch priority of a bridge group:

	Command	Purpose
Step 1	`ML_Series# configure terminal`	Enters the global configuration mode.
Step 2	`ML_Series(config)# bridge` *bridge-group-number* `priority` *priority-number*	Configures the switch priority of a bridge group. For *priority*, the range is 0 to 61440 in increments of 4096; the default is 32768. The lower the number, the more likely the switch will be chosen as the root switch. The value entered is rounded to the lower multiple of 4096. The actual number is computed by adding this number to the bridge group number.
Step 3	`ML_Series(config)# end`	Return to the privileged EXEC mode.

To return the switch to its default setting, use the **no bridge** *bridge-group-number* **priority** *priority-number* command.

Configuring the Hello Time

Change the hello time to configure the interval between the generation of configuration messages by the root switch.

Beginning in privileged EXEC mode, follow these steps to configure the hello time of a bridge group:

	Command	Purpose
Step 1	`ML_Series# configure terminal`	Enters global configuration mode.
Step 2	`ML_Series(config)# bridge` *bridge-group-number* `hello-time` *seconds*	Configures the hello time of a bridge group. The hello time is the interval between the generation of configuration messages by the root switch. These messages mean that the switch is alive. For *seconds*, the range is 1 to 10; the default is 2.
Step 3	`ML_Series(config)# end`	Returns to privileged EXEC mode.

To return the switch to its default setting, use the **no bridge** *bridge-group-number* **hello-time** *seconds* command. The number for *second*s should be the same number as configured in the original command.

Configuring the Forwarding-Delay Time for a Bridge Group

Beginning in privileged EXEC mode, follow these steps to configure the forwarding-delay time for a bridge group:

	Command	Purpose
Step 1	`ML_Series# configure terminal`	Enters global configuration mode.
Step 2	`ML_Series(config)# bridge` *bridge-group-number* `forward-time` *seconds*	Configures the forward time of a VLAN. The forward delay is the number of seconds a port waits before changing from its spanning-tree learning and listening states to the forwarding state. For *seconds*, the range is 4 to 200; the default is 15.
Step 3	`ML_Series(config)# end`	Returns to privileged EXEC mode.

To return the switch to its default setting, use the **no bridge** *bridge-group-number* **forward-time** *seconds* command. The number for *second*s should be the same number as configured in the original command.

Configuring the Maximum-Aging Time for a Bridge Group

Beginning in privileged EXEC mode, follow these steps to configure the maximum-aging time for a bridge group:

	Command	Purpose
Step 1	`ML_Series# configure terminal`	Enters global configuration mode.
Step 2	`ML_Series(config)# bridge` *bridge-group-number* `max-age` *seconds*	Configures the maximum-aging time of a bridge group. The maximum-aging time is the number of seconds a switch waits without receiving spanning-tree configuration messages before attempting a reconfiguration. For *seconds*, the range is 6 to 200; the default is 20.
Step 3	`ML_Series(config)# end`	Returns to privileged EXEC mode.

To return the switch to its default setting, use the **no bridge** *bridge-group-number* **max-age** *seconds* command. The number for *second*s should be the same number as configured in the original command.

Verifying and Monitoring STP and RSTP Status

To display the STP or RSTP status, use one or more of the privileged EXEC commands in Table 6-6.

Table 6-6 Commands for Displaying Spanning-Tree Status

Command	Purpose
ML_Series# **show spanning-tree**	Displays detailed STP or RSTP information.
ML_Series# **show spanning-tree brief**	Displays brief summary of STP or RSTP information.
ML_Series# **show spanning-tree interface** *interface-id*	Displays STP or RSTP information for the specified interface.
ML_Series# **show spanning-tree summary** [**totals**]	Displays a summary of port states or displays the total lines of the STP or RSTP state section.

Note The **show spanning-tree interface** *interface-id* privileged EXEC command displays information only if the port is in a link-up operative state. Otherwise, you can use the **show running-config interface** privileged EXEC command to confirm the configuration.

Examples of the **show spanning-tree** privileged EXEC commands are shown here:

Example 6-1 show spanning-tree Commands

```
ML_Series# show spanning-tree brief

Bridge group 1 is executing the rstp compatible Spanning Tree protocol
  Bridge Identifier has priority 32768, sysid 1, address 000b.fcfa.339e
  Configured hello time 2, max age 20, forward delay 15
  We are the root of the spanning tree
  Topology change flag not set, detected flag not set
  Number of topology changes 1 last change occurred 1w1d ago
          from POS0.1
  Times:  hold 1, topology change 35, notification 2
          hello 2, max age 20, forward delay 15
  Timers: hello 0, topology change 0, notification 0, aging 300

 Port 3 (FastEthernet0) of Bridge group 1 is designated disabled
   Port path cost 19, Port priority 128, Port Identifier 128.3.
   Designated root has priority 32769, address 000b.fcfa.339e
   Designated bridge has priority 32769, address 000b.fcfa.339e
   Designated port id is 128.3, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 0
   Link type is point-to-point by default
   BPDU: sent 0, received 0

ML_Series# show spanning-tree interface fastethernet 0
Port 3 (FastEthernet0) of Bridge group 1 is designated disabled
   Port path cost 19, Port priority 128, Port Identifier 128.3.
   Designated root has priority 32769, address 000b.fcfa.339e
   Designated bridge has priority 32769, address 000b.fcfa.339e
   Designated port id is 128.3, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 0
   Link type is point-to-point by default
   BPDU: sent 0, received 0

ML_Series# show spanning-tree summary totals
Switch is in pvst mode
Root bridge for: Bridge group 1-Bridge group 8
```

```
Name                   Blocking Listening Learning Forwarding STP Active
---------------------- -------- --------- -------- ---------- ----------
8 bridges                     8         0        0          0         16
```

CHAPTER 7

Configuring VLANs on the ML-Series Card

This chapter describes VLAN configurations for the ML-Series card. It describes how to configure IEEE 802.1Q VLAN encapsulation. For more information about the Cisco IOS commands used in this chapter, refer to the *Cisco IOS Command Reference* publication.

This chapter contains the following major sections:

Note Configuring VLANs is optional. Complete general interface configurations before proceeding with configuring VLANs as an optional step.

Understanding VLANs

VLANs enable network managers to group users logically rather than by physical location. A VLAN is an emulation of a standard LAN that allows secure intragroup data transfer and communication to occur without the traditional restraints placed on the network. It can also be considered a broadcast domain that is set up within a switch. With VLANs, switches can support more than one subnet (or VLAN) on each switch and give routers and switches the opportunity to support multiple subnets on a single physical link. A group of devices that belong to the same VLAN, but are part of different LAN segments, are configured to communicate as if they were part of the same LAN segment.

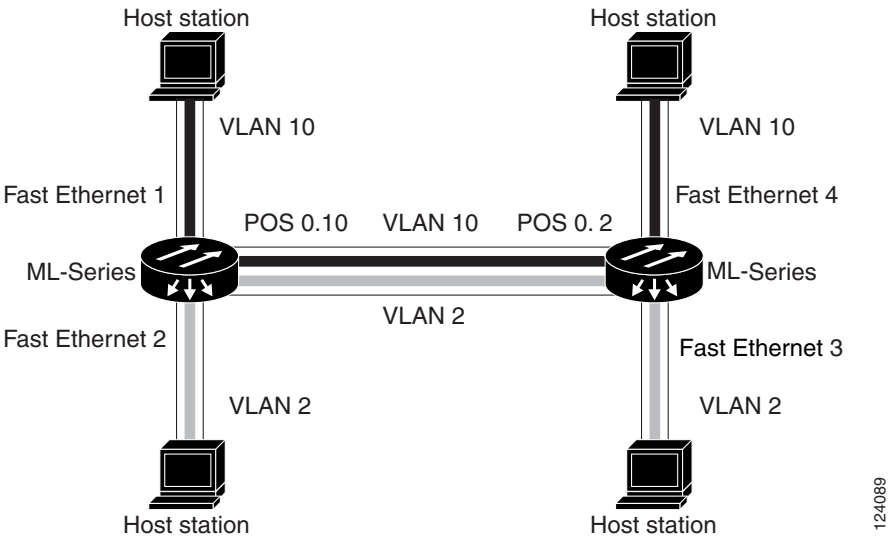
VLANs enable efficient traffic separation and provide excellent bandwidth utilization. VLANs also alleviate scaling issues by logically segmenting the physical LAN structure into different subnetworks so that packets are switched only between ports within the same VLAN. This can be very useful for security, broadcast containment, and accounting.

ML-Series software supports port-based VLANs and VLAN trunk ports, which are ports that carry the traffic of multiple VLANs. Each frame transmitted on a trunk link is tagged as belonging to only one VLAN.

ML-Series software supports VLAN frame encapsulation through the IEEE 802.1Q standard. The Cisco Inter-Switch Link (ISL) VLAN frame encapsulation is not supported. ISL frames are broadcast at Layer 2 or dropped at Layer 3.

ML-Series switching supports up to 254 VLAN subinterfaces per interface. A maximum of 255 logical VLANs can be bridged per card (limited by the number of bridge-groups). Each VLAN subinterface can be configured for any VLAN ID in the full 1 to 4095 range. Figure 7-1 shows a network topology in which two VLANs span two ONS 15310-CLs with ML-Series cards.

Figure 7-1 VLANs Spanning Devices in a Network



Configuring IEEE 802.1Q VLAN Encapsulation

You can configure IEEE 802.1Q VLAN encapsulation on either type of ML-Series card interfaces, Ethernet or Packet over SONET/SDH (POS). VLAN encapsulation is not supported on POS interfaces configured with HDLC encapsulation.

The native VLAN is always VLAN ID 1 on ML-Series cards. Frames on the native VLAN are normally transmitted and received untagged. On an trunk port, all frames from VLANs other than the native VLAN are transmitted and received tagged.

To configure VLANs using IEEE 802.1Q VLAN encapsulation, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	`ML_Series(config)# bridge` *bridge-group-number* `protocol` *type*	Assigns a bridge group (VLAN) number and define the appropriate spanning tree type.
Step 2	`ML_Series(config)# interface` *type number*	Enters interface configuration mode to configure the interface.
Step 3	`ML_Series(config)# interface` *type number.subinterface-number*	Enters subinterface configuration mode to configure the subinterface.
Step 4	`ML_Series(config-subif)# encap dot1q` *vlan-id*	Sets the encapsulation format on the VLAN to IEEE 802.1Q.
Step 5	`ML_Series(config-subif)# bridge-group` *bridge-group-number*	Assigns a network interface to a bridge group.

	Command	Purpose
Step 6	`ML_Series(config-subif)# end`	Returns to privileged EXEC mode.
Step 7	`ML_Series# copy running-config startup-config`	(Optional) Saves your configuration changes to NVRAM.

Note In a bridge group on the ML-Series card, the VLAN ID does not have to be uniform across interfaces that belong to that bridge group. For example, a bridge-group can connect from a VLAN ID subinterface to a subinterface with a different VLAN ID, and then frames entering with one VLAN ID can be changed to exit with a different VLAN ID. This is know as VLAN translation.

Note IP routing is enabled by default. To enable bridging, enter the **no ip routing** or **bridge IRB** command.

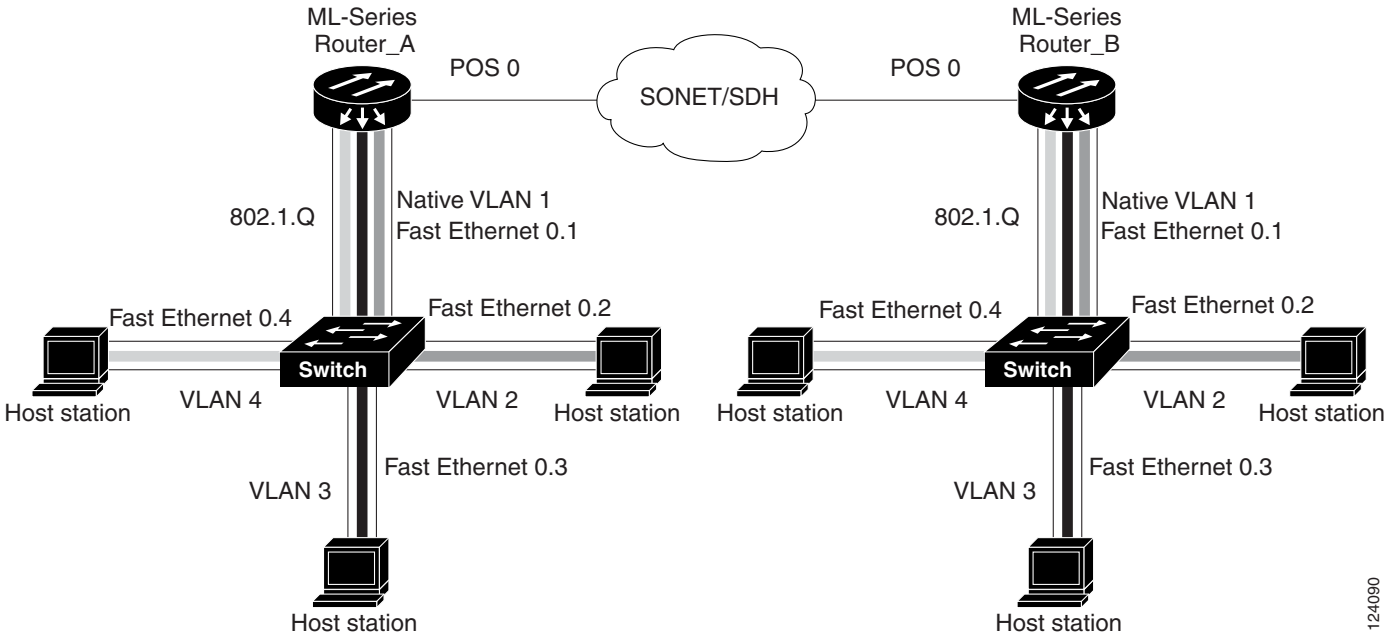
Note Native VLAN frames transmitted on the interface are normally untagged. All untagged frames received on the interface are associated with the native VLAN, which is always VLAN 1. Use the command **encapsulation dot1q 1 native**.

IEEE 802.1Q VLAN Configuration

The VLAN configuration example for the ML-Series card shown in Figure 7-2 depicts the following VLANs:

- Fast Ethernet subinterface 0.1 is in the IEEE 802.1Q native VLAN 1.
- Fast Ethernet subinterface 0.2 is in the IEEE 802.1Q VLAN 2.
- Fast Ethernet subinterface 0.3 is in the IEEE 802.1Q VLAN 3.
- Fast Ethernet subinterface 0.4 is in the IEEE 802.1Q VLAN 4.

Figure 7-2 Bridging IEEE 802.1Q VLANs



Example 7-1 shows how to configure VLANs for IEEE 802.1Q VLAN encapsulation. Use this configuration for both ML_Series A and ML_Series B.

Example 7-1 Configure VLANs for IEEE 8021Q VLAN Encapsulation

```
no ip routing
bridge 1 protocol ieee
bridge 2 protocol ieee
bridge 3 protocol ieee
bridge 4 protocol ieee
!
!
interface FastEthernet0
!
interface FastEthernet0.1
 encapsulation dot1Q 1 native
 bridge-group 1
!
interface FastEthernet0.2
 encapsulation dot1Q 2
 bridge-group 2
!
interface FastEthernet0.3
 encapsulation dot1Q 3
 bridge-group 3
!
interface FastEthernet0.4
 encapsulation dot1Q 4
 bridge-group 4
!
interface POS0
!
interface POS0.1
 encapsulation dot1Q 1 native
 bridge-group 1
```

```
!
interface POS0.2
 encapsulation dot1Q 2
 bridge-group 2
!
interface POS0.3
 encapsulation dot1Q 3
 bridge-group 3
!
interface POS0.4
 encapsulation dot1Q 4
 bridge-group 4
```

Monitoring and Verifying VLAN Operation

After the VLANs are configured on the ML-Series card, you can monitor their operation by entering the privileged EXEC command **show vlans** [*vlan-id*] (Example 7-2). This command displays information on all configured VLANs or on a specific VLAN (by VLAN ID number).

Example 7-2 Output for show vlans Command

```
ML-Series# show vlans 1

Virtual LAN ID:  1 (IEEE 802.1Q Encapsulation)

   vLAN Trunk Interface:   POS0.1

 This is configured as native Vlan for the following interface(s) :
POS0

   Protocols Configured:   Address:              Received:        Transmitted:
        Bridging        Bridge Group 1                  0                   0
```

CHAPTER 8

Configuring IEEE 802.1Q Tunneling and Layer 2 Protocol Tunneling on the ML-Series Card

Virtual private networks (VPNs) provide enterprise-scale connectivity on a shared infrastructure, often Ethernet-based, with the same security, prioritization, reliability, and manageability requirements of private networks. Tunneling is a feature designed for service providers who carry traffic of multiple customers across their networks and are required to maintain the VLAN and Layer 2 protocol configurations of each customer without impacting the traffic of other customers. The ML-Series cards support IEEE 802.1Q tunneling (QinQ) and Layer 2 protocol tunneling.

This chapter contains the following sections:

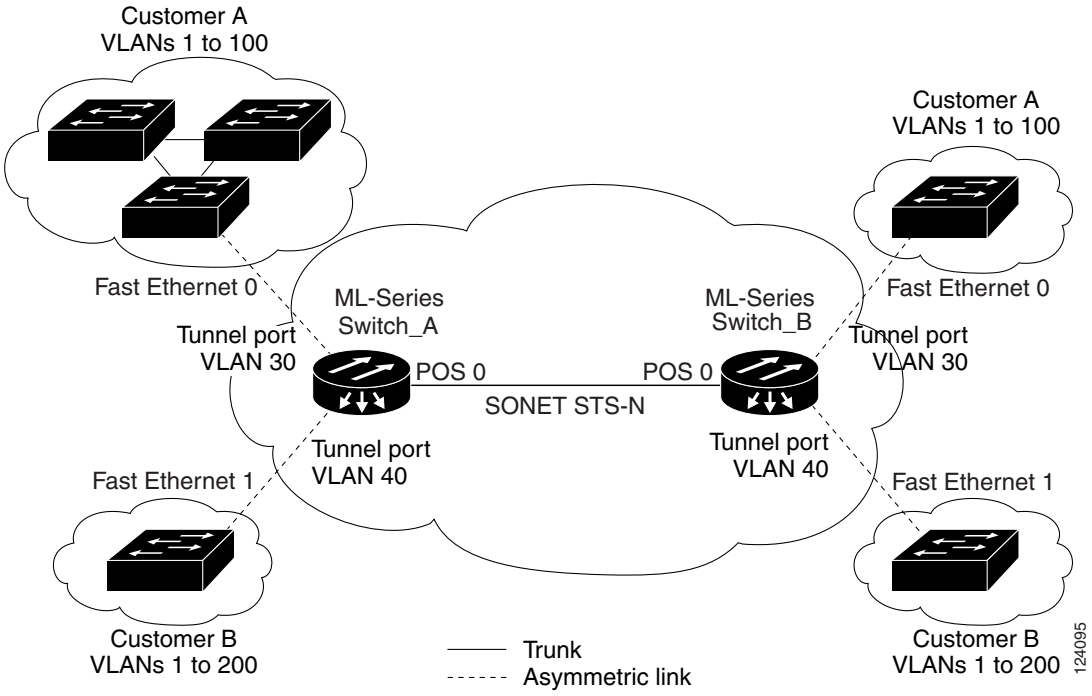
Understanding IEEE 802.1Q Tunneling

Business customers of service providers often have specific requirements for VLAN IDs and the number of VLANs to be supported. The VLAN ranges required by different customers in the same service-provider network might overlap, and traffic of customers through the infrastructure might be mixed. Assigning a unique range of VLAN IDs to each customer would restrict customer configurations and could easily exceed the IEEE 802.1Q specification VLAN limit of 4096.

Using the IEEE 802.1Q tunneling (QinQ) feature, service providers can use a single VLAN to support customers who have multiple VLANs. Customer VLAN IDs are preserved and traffic from different customers is segregated within the service-provider infrastructure even when they appear to be on the same VLAN. The IEEE 802.1Q tunneling expands VLAN space by using a VLAN-in-VLAN hierarchy and tagging the tagged packets. A port configured to support IEEE 802.1Q tunneling is called a tunnel port. When you configure tunneling, you assign a tunnel port to a VLAN that is dedicated to tunneling. Each customer requires a separate VLAN, but that VLAN supports all of the customer's VLANs.

Customer traffic tagged in the normal way with appropriate VLAN IDs comes from an IEEE 802.1Q trunk port on the customer device and into a tunnel port on the ML-Series card. The link between the customer device and the ML-Series card is an asymmetric link because one end is configured as an IEEE 802.1Q trunk port and the other end is configured as a tunnel port. You assign the tunnel port interface to an access VLAN ID unique to each customer (Figure 8-1).

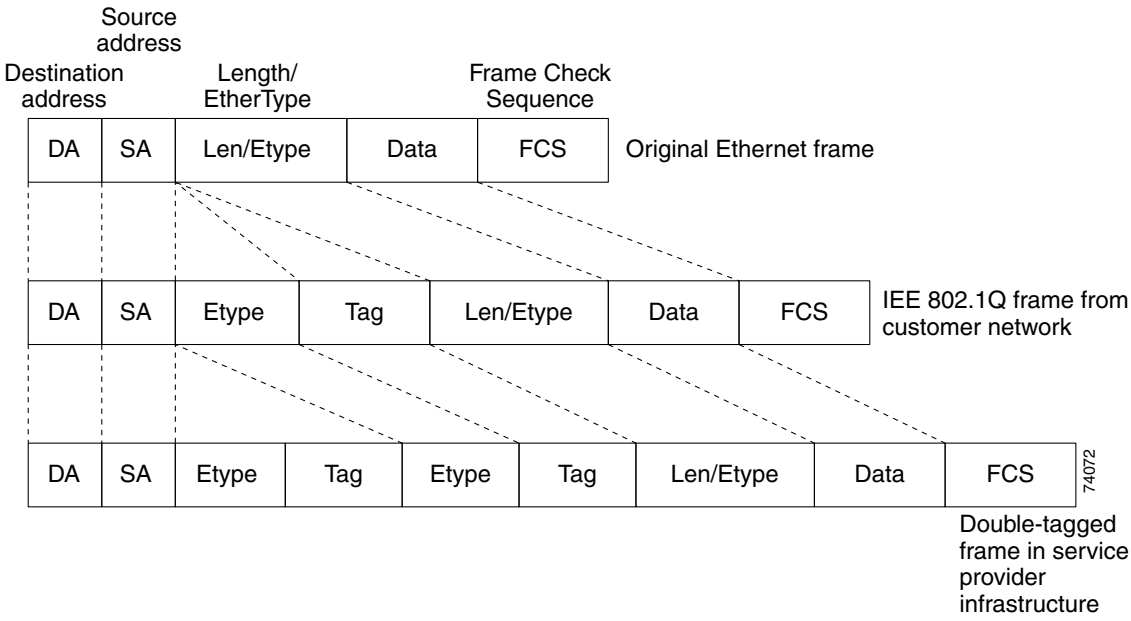
Figure 8-1 IEEE 802.1Q Tunnel Ports in a Service-Provider Network



Packets coming from the customer trunk port into the tunnel port on the ML-Series card are normally IEEE 802.1Q-tagged with an appropriate VLAN ID. The tagged packets remain intact inside the ML-Series card and, when they exit the trunk port into the service provider network, are encapsulated with another layer of an IEEE 802.1Q tag (called the *metro tag*) that contains the VLAN ID unique to the customer. The original IEEE 802.1Q tag from the customer is preserved in the encapsulated packet. Therefore, packets entering the service-provider infrastructure are double-tagged, with the outer tag containing the customer's access VLAN ID, and the inner VLAN ID being the VLAN of the incoming traffic.

When the double-tagged packet enters another trunk port in a service provider ML-Series card, the outer tag is stripped as the packet is processed inside the switch. When the packet exits another trunk port on the same core switch, the same metro tag is again added to the packet. Figure 8-2 shows the structure of the double-tagged packet.

Figure 8-2 Normal, IEEE 802.1Q, and IEEE 802.1Q-Tunneled Ethernet Packet Formats



When the packet enters the trunk port of the service-provider egress switch, the outer tag is again stripped as the packet is processed internally on the switch. However, the metro tag is not added when it is sent out the tunnel port on the edge switch into the customer network, and the packet is sent as a normal IEEE 802.1Q-tagged frame to preserve the original VLAN numbers in the customer network.

In Figure 8-1 on page 8-2, Customer A was assigned VLAN 30, and Customer B was assigned VLAN 40. Packets entering the ML-Series card tunnel ports with IEEE 802.1Q tags are double-tagged when they enter the service-provider network, with the outer tag containing VLAN ID 30 or 40, appropriately, and the inner tag containing the original VLAN number, for example, VLAN 100. Even if both Customers A and B have VLAN 100 in their networks, the traffic remains segregated within the service-provider network because the outer tag is different. With IEEE 802.1Q tunneling, each customer controls its own VLAN numbering space, which is independent of the VLAN numbering space used by other customers and the VLAN numbering space used by the service-provider network.

At the outbound tunnel port, the original VLAN numbers on the customer's network are recovered. If the traffic coming from a customer network is not tagged (native VLAN frames), these packets are bridged or routed as if they were normal packets, and the metro tag is added (as a single-level tag) when they exit toward the service provider network.

If the native VLAN (VLAN 1) is used in the service provider network as a metro tag, this tag must always be added to the customer traffic, even though the native VLAN ID is not normally added to transmitted frames. If the VLAN 1 metro tag is not added on frames entering the service provider network, then the customer VLAN tag appears to be the metro tag, with disastrous results. The global configuration **vlan dot1q tag native** command must be used to prevent this by forcing a tag to be added to VLAN 1. Avoiding the use of VLAN 1 as a metro tag transporting customer traffic is recommended to reduce the risk of misconfiguration. A best practice is to use VLAN 1 as a private management VLAN in the service provider network.

The IEEE 802.1Q class of service (COS) priority field on the added metro tag is set to zero by default, but can be modified by input or output policy maps.

Configuring IEEE 802.1Q Tunneling

This section includes the following information about configuring IEEE 802.1Q tunneling:

Note By default, IEEE 802.1Q tunneling is not configured on the ML-Series.

IEEE 802.1Q Tunneling and Compatibility with Other Features

Although IEEE 802.1Q tunneling works well for Layer 2 packet switching, there are incompatibilities with some Layer 2 features and with Layer 3 switching:

- A tunnel port cannot be a routed port.
- Tunnel ports do not support IP access control lists (ACLs).
- Layer 3 quality of service (QoS) ACLs and other QoS features related to Layer 3 information are not supported on tunnel ports. MAC-based QoS is supported on tunnel ports.
- EtherChannel port groups are compatible with tunnel ports as long as the IEEE 802.1Q configuration is consistent within an EtherChannel port group.
- Port Aggregation Protocol (PAgP) and Unidirectional Link Detection (UDLD) Protocol are not supported on IEEE 802.1Q tunnel ports.
- Dynamic Trunking Protocol (DTP) is not compatible with IEEE 802.1Q tunneling because you must manually configure asymmetric links with tunnel ports and trunk ports.
- Loopback detection is supported on IEEE 802.1Q tunnel ports.
- When a port is configured as an IEEE 802.1Q tunnel port, spanning tree bridge protocol data unit (BPDU) filtering is automatically disabled on the interface.

Configuring an IEEE 802.1Q Tunneling Port

Beginning in privileged EXEC mode, follow these steps to configure a port as an IEEE 802.1Q tunnel port:

	Command	Purpose
Step 1	`ML_Series# configure terminal`	Enters global configuration mode.
Step 2	`ML_Series(config)# bridge` *bridge-number* `protocol` *bridge-protocol*	Creates a bridge number and specifies a protocol.
Step 3	`ML_Series(config)# interface fastethernet` *number*	Enters the interface configuration mode and the interface to be configured as a tunnel port. This should be the edge port in the service-provider network that connects to the customer switch. Valid interfaces include physical interfaces and port-channel logical interfaces.

	Command	Purpose
Step 4	`ML_Series(config-if)# bridge-group` *vlan-number*	Assigns the tunnel port to a VLAN. All traffic from the port (tagged and untagged) will be switched based on this bridge-group. Other members of the bridge-group should be VLAN subinterfaces on a provider trunk interface.
Step 5	`ML_Series(config-if)# mode dot1q-tunnel`	Sets the interface as an IEEE 802.1Q tunnel port to enable QinQ.
Step 6	`ML_Series(config)# end`	Returns to privileged EXEC mode.
Step 7	`ML_Series# show dot1q-tunnel`	Displays the tunnel ports on the switch.
Step 8	`ML_Series# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Note The VLAN ID (VID) range of 2 to 4095 is recommended for IEEE 802.1Q tunneling on the ML-Series card.

Note If VID 1 is required to be used as a metro tag, use the **VLAN dot1Q tag native** global configuration command.

Use the **no mode dot1q-tunnel** interface configuration command to remove the IEEE 802.1Q tunnel from the interface.

IEEE 802.1Q Example

The following examples show how to configure the example in Figure 8-1 on page 8-2. Example 8-1 applies to ML_Series A, and Example 8-2 applies to ML_Series B.

Example 8-1 ML_Series A Configuration

```
no ip routing
bridge 30 protocol ieee
bridge 40 protocol ieee
!
!
interface FastEthernet0
no ip routing
mode dot1q-tunnel
 bridge-group 30
!
interface FastEthernet1
mode dot1q-tunnel
 bridge-group 40
!
interface POS0
!
interface POS0.1
 encapsulation dot1Q 30
 bridge-group 30
!
interface POS0.2
 encapsulation dot1Q 40
```

```
 bridge-group 40
```

Example 8-2 ML_Series B Configuration

```
no ip routing
bridge 30 protocol ieee
bridge 40 protocol ieee
!
!
interface FastEthernet0
no ip routing
mode dot1q-tunnel
 bridge-group 30
!
interface FastEthernet1
mode dot1q-tunnel
 bridge-group 40
!
interface POS0
!
interface POS0.1
 encapsulation dot1Q 30
 bridge-group 30
!
interface POS0.2
 encapsulation dot1Q 40
 bridge-group 40
```

Understanding VLAN-Transparent and VLAN-Specific Services

The ML-Series card supports combining VLAN-transparent services and one or more VLAN-specific services on the same port. All of these VLAN-transparent and VLAN-specific services can be point-to-point or multipoint-to-multipoint.

This allows a service provider to combine a VLAN-transparent service, such as IEEE 802.1Q tunneling (QinQ), with VLAN-specific services, such as bridging specific VLANs, on the same customer port. For example, one customer VLAN can connect to Internet access and the other customer VLANs can be tunneled over a single provider VLAN to another customer site, all over a single port at each site. Table 8-1 outlines the differences between VLAN-transparent and VLAN-specific services.

Table 8-1 VLAN-Transparent Service Versus VLAN-Specific Services

VLAN-Transparent Services	VLAN-Specific Services
Bridging only	Bridging or routing
One service per port	Up to 254 VLAN-specific services per port
Applies indiscriminately to all VLANs on the physical interface	Applies only to specified VLANs

Note VLAN-transparent service is also referred to as Ethernet Wire Service (EWS). VLAN-specific service is also referred to as QinQ tunneling trunk UNI in Metro Ethernet terminology.

A VLAN-specific service on a subinterface coexists with the VLAN-transparent service, often IEEE 802.1Q tunneling, on a physical interface. VLANs configured for a VLAN-transparent service and a VLAN-specific service follow the VLAN-specific service configuration. If you need to configure IEEE 802.1Q tunneling, configure this VLAN-transparent service in the normal manner (see the "Configuring IEEE 802.1Q Tunneling" section on page 8-4).

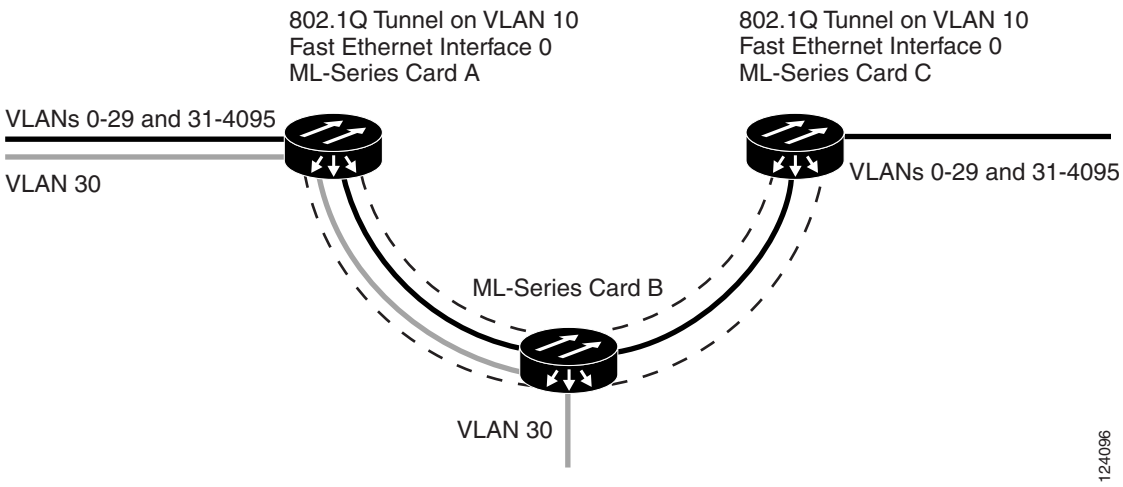
A VLAN-specific service can be any service normally applicable to a VLAN. To configure an ERMS VLAN-specific service, configure the service in the normal manner.

VLAN-Transparent and VLAN-Specific Services Configuration Example

In this example, the Fast Ethernet interface 0 on both the ML-Series card A and ML-Series card C are the trunk ports in an IEEE 802.1Q tunnel, which is a VLAN-transparent service. VLAN 10 is used for the VLAN-transparent service, which would normally transport all customer VLANs on the ML-Series card A's Fast Ethernet interface 0. All unspecified VLANs and VLAN 1 would also be tunneled across VLAN 10.

VLAN 30 is prevented from entering the VLAN-transparent service and is instead forwarded on a specific-VLAN service, bridging Fast Ethernet interface 0 on ML-Series card A and Fast Ethernet interface 0 on ML-Series card B. Figure 8-3 is a reference for Example 8-3, Example 8-4 on page 8-8, and Example 8-5 on page 8-8.

Figure 8-3 ERMS Example



Example 8-3 applies to ML-Series card A.

Example 8-3 ML-Series Card A Configuration

```
hostname ML-A
no ip routing
bridge 10 protocol rstp
bridge 30 protocol ieee
!
!
interface FastEthernet0
```

```
       mode dot1q-tunnel
    bridge-group 10
     bridge-group 10 spanning-disabled
!
interface FastEthernet0.3
 encapsulation dot1Q 30
bridge-group 30
!
interface POS0
!
interface POS0.1
 encapsulation dot1Q 10
bridge-group 10
!
interface POS0.3
 encapsulation dot1Q 30
bridge-group 30
```

Example 8-4 applies to ML-Series card B.

Example 8-4 ML-Series Card B Configuration

```
hostname ML-B
!
bridge 10 protocol rstp
bridge 30 protocol ieee
!
!
interface FastEthernet0
!
interface FastEthernet0.3
 encapsulation dot1Q 30
 bridge-group 30
!
interface FastEthernet1
shutdown
!
interface POS0.1
 encapsulation dot1Q 10
 bridge-group 10
!
interface POS0.3
 encapsulation dot1Q 30
 bridge-group 30
!
interface POS1.1
 encapsulation dot1Q 10
 bridge-group 10
!
interface POS1.3
 encapsulation dot1Q 30
 bridge-group 30
```

Example 8-5 applies to ML-Series card C.

Example 8-5 ML-Series Card C Configuration

```
hostname ML-C
bridge 10 protocol rstp
!
!
interface FastEthernet0
```

```
    no ip address
    no ip route-cache
    mode dot1q-tunnel
    bridge-group 10
     bridge-group 10 spanning-disabled
!
interface POS0.1
 encapsulation dot1Q 10
 no ip route-cache
 bridge-group 10
```

Understanding Layer 2 Protocol Tunneling

Customers at different sites connected across a service-provider network need to run various Layer 2 protocols to scale their topology to include all remote sites, as well as the local sites. Spanning Tree Protocol (STP) must run properly, and every VLAN should build a proper spanning tree that includes the local site and all remote sites across the service-provider infrastructure. Cisco Discovery Protocol (CDP) must discover neighboring Cisco devices from local and remote sites. VLAN Trunking Protocol (VTP) must provide consistent VLAN configuration throughout all sites in the customer network.

When protocol tunneling is enabled, edge switches on the inbound side of the service-provider infrastructure encapsulate Layer 2 protocol packets with a special MAC address and send them across the service-provider network. Core switches in the network do not process these packets, but forward them as normal packets. CDP, STP, or VTP Layer 2 protocol data units (PDUs) cross the service-provider infrastructure and are delivered to customer switches on the outbound side of the service-provider network. Identical packets are received by all customer ports on the same VLANs with the following results:

- Users on each of a customer's sites are able to properly run STP and every VLAN can build a correct spanning tree based on parameters from all sites and not just from the local site.
- CDP discovers and shows information about the other Cisco devices connected through the service-provider network.
- VTP provides consistent VLAN configuration throughout the customer network, propagating through the service provider to all switches.

Layer 2 protocol tunneling can be used independently or to enhance IEEE 802.1Q tunneling. If protocol tunneling is not enabled on IEEE 802.1Q tunneling ports or on specific VLANs, remote switches at the receiving end of the service-provider network do not receive the PDUs and cannot properly run STP, CDP, and VTP. When protocol tunneling is enabled, Layer 2 protocols within each customer's network are totally separate from those running within the service-provider network. Customer switches on different sites that send traffic through the service-provider network with IEEE 802.1Q tunneling achieve complete knowledge of the customer's VLAN. If IEEE 802.1Q tunneling is not used, you can still enable Layer 2 protocol tunneling by connecting to the customer switch through access ports and enabling tunneling on the service-provider access port.

Configuring Layer 2 Protocol Tunneling

Layer 2 protocol tunneling (by protocol) is enabled on the tunnel ports or on specific tunnel VLANs that are connected to the customer by the edge switches of the service-provider network. ML-Series card tunnel ports are connected to customer IEEE 802.1Q trunk ports. The ML-Series card supports Layer 2

protocol tunneling for CDP, STP, and VTP at the interface and subinterface level. Multiple STP (MSTP) tunneling support is achieved through subinterface protocol tunneling. The ML-Series cards connected to the customer switch perform the tunneling process.

When the Layer 2 PDUs that entered the inbound ML-Series switch through the tunnel port exit the switch through the trunk port into the service-provider network, the switch overwrites the customer PDU-destination MAC address with a well-known Cisco proprietary multicast address (01-00-0c-cd-cd-d0). If IEEE 802.1Q tunneling is enabled, packets are also double-tagged; the outer tag is the customer metro tag and the inner tag is the customer VLAN tag. The core switches ignore the inner tags and forward the packet to all trunk ports in the same metro VLAN. The ML-Series switches on the outbound side restore the proper Layer 2 protocol and MAC address information and forward the packets. Therefore, the Layer 2 PDUs are kept intact and delivered across the service-provider infrastructure to the other side of the customer network.

This section contains the following information about configuring Layer 2 protocol tunneling:

Default Layer 2 Protocol Tunneling Configuration

Table 8-2 shows the default Layer 2 protocol tunneling configuration.

Table 8-2 Default Layer 2 Protocol Tunneling Configuration

Feature	Default Setting
Layer 2 protocol tunneling	Disabled for CDP, STP, and VTP.
Class of service (CoS) value	If a CoS value is configured on the interface for data packets, that value is the default used for Layer 2 PDUs. If none is configured, there is no default. This allows existing CoS values to be maintained, unless the user configures otherwise.

Layer 2 Protocol Tunneling Configuration Guidelines

These are some configuration guidelines and operating characteristics of Layer 2 protocol tunneling:

- The ML-Series card supports Per-VLAN Protocol Tunneling (PVPT), which allows protocol tunneling to be configured and run on a specific subinterface (VLAN). PVPT configuration is done at the subinterface level.
- PVPT should be configured on VLANs that carry multi-session transport (MST) BPDUs on the connected devices.
- The ML-Series card supports tunneling of CDP and STP (including MSTP and VTP protocols). Protocol tunneling is disabled by default but can be enabled for the individual protocols on IEEE 802.1Q tunnel ports or on specific VLANs.

- Tunneling is not supported on trunk ports. If you enter the **l2protocol-tunnel** interface configuration command on a trunk port, the command is accepted, but Layer 2 tunneling does not take effect unless you change the port to a tunnel port.
- EtherChannel port groups are compatible with tunnel ports as long as the IEEE 802.1Q configuration is configured within an EtherChannel port group.
- If an encapsulated PDU (with the proprietary destination MAC address) is received from a tunnel port or access port with Layer 2 tunneling enabled, the tunnel port is shut down to prevent loops.
- Only decapsulated PDUs are forwarded to the customer network. The spanning tree instance running on the service-provider network does not forward BPDUs to tunnel ports. No CDP packets are forwarded from tunnel ports.
- Because tunneled PDUs (especially STP BPDUs) must be delivered to all remote sites for the customer virtual network to operate properly, you can give PDUs higher priority within the service-provider network than data packets received from the same tunnel port. By default, the PDUs use the same CoS value as data packets.
- Protocol tunneling has to be configured symmetrically at both the ingress and egress point. For example, if you configure the entry point to tunnel STP, CDP, and VTP, then you must configure the egress point in the same way.

Configuring Layer 2 Tunneling on a Port

Beginning in privileged EXEC mode, follow these steps to configure a port as a Layer 2 tunnel port:

	Command	Purpose
Step 1	`ML_Series# configuration terminal`	Enters global configuration mode.
Step 2	`ML_Series(config)# bridge` *bridge-group-number* `protocol` *type*	Creates a bridge group number and specifies a protocol.
Step 3	`ML_Series(config)# l2protocol-tunnel cos` ***cos-value***	Associates a CoS value with the Layer 2 tunneling port. Valid numbers for *cos-value* range from 0 to 7.
Step 4	`ML_Series(config)# interface` ***type*** *number*	Enters interface configuration mode for the interface to be configured as a tunnel port.
Step 5	`ML_Series(config-if)# bridge-group` *bridge-group-number*	Assigns a bridge group to the interface.
Step 6	`ML_Series(config-if)# mode dot1q tunnel`	Sets the interface as an IEEE 802.1Q tunnel VLAN.
Step 7	`ML_Series(config-if)# l2protocol-tunnel {all | cdp | stp | vtp}`	Sets the interface as a Layer 2 protocol tunnel port and enables all three protocols or specifically enables CDP, STP, or VTP. These protocols are off by default.
Step 8	`ML_Series(config-if)# end`	Returns to privileged EXEC mode.
Step 9	`ML_Series# show dot1q-tunnel`	Displays the tunnel ports on the switch.
Step 10	`ML_Series# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Layer 2 Tunneling Per-VLAN

Beginning in privileged EXEC mode, follow these steps to configure a VLAN as a Layer 2 tunnel VLAN:

	Command	Purpose
Step 1	`ML_Series# configuration terminal`	Enters global configuration mode.
Step 2	`ML_Series(config)# bridge` *bridge-group-number* `protocol` *type*	Creates a bridge group number and specifies a protocol.
Step 3	`ML_Series(config)# l2protocol-tunnel cos` ***cos-value***	Associates a CoS value with the Layer 2 tunneling VLAN. Valid numbers for *cos-value* range from 0 to 7.
Step 4	`ML_Series(config)# interface` ***type*** *number.subinterface-number*	Enters subinterface configuration mode and the subinterface to be configured as a tunnel VLAN.
Step 5	`ML_Series(config-subif)# encapsulation dot1q` ***bridge-group-number***	Sets the subinterface as an IEEE 802.1Q tunnel VLAN.
Step 6	`ML_Series(config-subif)# bridge-group` ***bridge-group-****number*	Specifies the default VLAN, which is used if the subinterface stops trunking. This VLAN ID is specific to the particular customer.
Step 7	`ML_Series(config-subif)# end`	Returns to privileged EXEC mode.
Step 8	`ML_Series# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring and Verifying Tunneling Status

Table 8-3 shows the privileged EXEC commands for monitoring and maintaining IEEE 802.1Q and Layer 2 protocol tunneling.

Table 8-3 Commands for Monitoring and Maintaining Tunneling

Command	Purpose
`show dot1q-tunnel`	Displays IEEE 802.1Q tunnel ports on the switch.
`show dot1q-tunnel interface` *interface-id*	Verifies if a specific interface is a tunnel port.
`show l2protocol-tunnel`	Displays information about Layer 2 protocol tunneling ports.
`show vlan dot1q tag native`	Displays IEEE 802.1Q tunnel information.

CHAPTER 9

Configuring Link Aggregation on the ML-Series Card

This chapter describes how to configure link aggregation for the ML-Series cards, both EtherChannel and packet-over-SONET (POS) channel. For additional information about the Cisco IOS commands used in this chapter, refer to the *Cisco IOS Command Reference* publication.

This chapter contains the following major sections:

Understanding Link Aggregation

The ML-Series card offers both EtherChannel and POS channel. Traditionally EtherChannel is a trunking technology that groups together multiple full-duplex IEEE 802.3 Ethernet interfaces to provide fault-tolerant high-speed links between switches, routers, and servers. EtherChannel forms a single higher bandwidth routing or bridging endpoint and was designed primarily for host-to-switch connectivity. The ML-Series card extends this link aggregation technology to bridged POS interfaces. POS channel is only supported with LEX encapsulation.

Link aggregation provides the following benefits:

- Logical aggregation of bandwidth
- Load balancing
- Fault tolerance

Port channel is a term for both POS channel and EtherChannel. The port channel interface is treated as a single logical interface although it consists of multiple interfaces. Each port channel interfaces consists of one type of interface, either Fast Ethernet or POS. You must perform all port channel configurations on the port channel (EtherChannel or POS channel) interface rather than on the individual member Ethernet or POS interfaces. You can create the port channel interface by entering the **interface port-channel** interface configuration command.

Port channel connections are fully compatible with IEEE 802.1Q trunking and routing technologies. IEEE 802.1Q trunking can carry multiple VLANs across a port channel.

Each ML100-FX supports up to four FECs plus an additional POS channel, a port channel made up of the two POS ports. A maximum of four Fast Ethernet ports can bundle into one Fast Ethernet Channel (FEC) and provide bandwidth scalability up to 400-Mbps full-duplex Fast Ethernet.

Caution The EtherChannel interface is the Layer 2/Layer 3 interface. Do not enable Layer 3 addresses on the physical interfaces. Do not assign bridge groups on the physical interfaces because doing so creates loops.

Caution Before a physical interface is removed from an EtherChannel (port channel) interface, the physical interface must be disabled. To disable a physical interface, use the **shutdown** command in interface configuration mode.

Note Link aggregation across multiple ML-Series cards is not supported.

Note Policing is not supported on port channel interfaces.

Note The ML-Series does not support the routing of Subnetwork Access Protocol (SNAP) or Inter-Switch Link (ISL) encapsulated frames.

Configuring Link Aggregation

You can configure an FEC or POS channel by creating an EtherChannel interface (port channel) and optionally assigning a network IP address.

Configuring Fast EtherChannel

All interfaces that are members of an FEC should have the same link parameters, such as duplex and speed.

To create an EtherChannel interface, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface port-channel channel-number`	Creates the EtherChannel interface.
Step 2	`Router(config-if)# ip address ip-address subnet-mask`	(Optional) Assigns an IP address and subnet mask to the EtherChannel interface.
Step 3	`Router(config-if)# end`	Exits to privileged EXEC mode.
Step 4	`Router# copy running-config startup-config`	(Optional) Saves configuration changes to NVRAM.

For information on other configuration tasks for the EtherChannel, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide*.

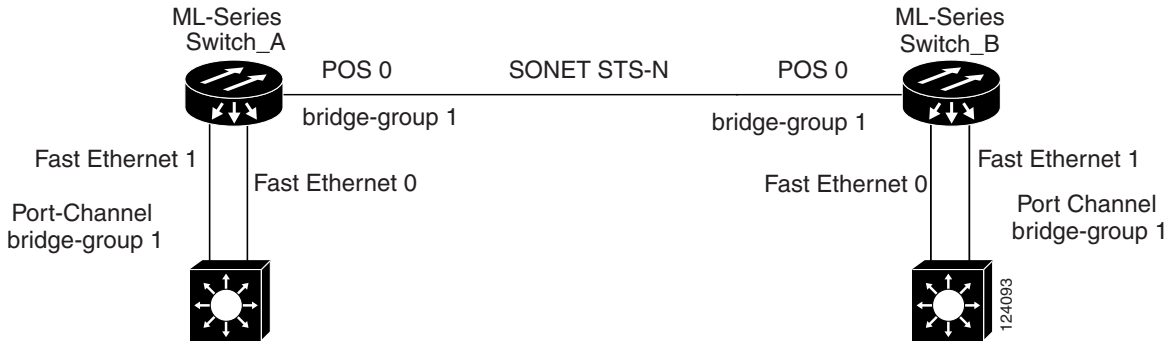
To assign Ethernet interfaces to the EtherChannel, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface fastethernet number`	Enters one of the interface configuration modes to configure the Fast Ethernet interface that you want to assign to the EtherChannel.
Step 2	`Router(config-if)# channel-group channel-number`	Assigns the Fast Ethernet interface to the EtherChannel. The channel number must be the same channel number you assigned to the EtherChannel interface.
Step 3	`Router(config-if)# end`	Exits to privileged EXEC mode.
Step 4	`Router# copy running-config startup-config`	(Optional) Saves configuration changes to NVRAM.

EtherChannel Configuration Example

Figure 9-1 shows an example of encapsulation over EtherChannel. The associated commands are provided in Example 9-1 and Example 9-2.

Figure 9-1 Encapsulation over EtherChannel Example



Example 9-1 ML_Series A Configuration

```
hostname Switch A
no ip routing
!
bridge 1 protocol ieee
!
interface Port-channel 1
bridge-group 1
 hold-queue 150 in
!
interface FastEthernet 0
channel-group 1
!
```

```
interface FastEthernet 1
channel-group 1
!
interface POS 0
bridge-group 1
```

Example 9-2 ML_Series B Configuration

```
hostname Switch B
no ip routing
!
bridge 1 protocol ieee
!
interface Port-channel 1
bridge-group 1
 hold-queue 150 in
!
interface FastEthernet 0
channel-group 1
!
interface FastEthernet 1
channel-group 1
!
interface POS 0
bridge-group 1
!
```

Configuring POS Channel

You can configure a POS channel by creating a POS channel interface (port channel) and optionally assigning an IP address. All POS interfaces that are members of a POS channel should have the same port properties and be on the same ML-Series card.

Note POS channel is only supported with G-Series card compatible (LEX) encapsulation.

To create a POS channel interface, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface port-channel` *channel-number*	Creates the POS channel interface. You can configure one POS channel on the ML-Series card.
Step 2	`Router(config-if)# ip address` *ip-address subnet-mask*	Assigns an IP address and subnet mask to the POS channel interface (required only for the Layer 3 POS channel).
Step 3	`Router(config-if)# end`	Exits to privileged EXEC mode.
Step 4	`Router# copy running-config startup-config`	(Optional) Saves configuration changes to NVRAM.

Caution The POS channel interface is the routed interface. Do not enable Layer 3 addresses on any physical interfaces. Do not assign bridge groups on any physical interfaces because doing so creates loops.

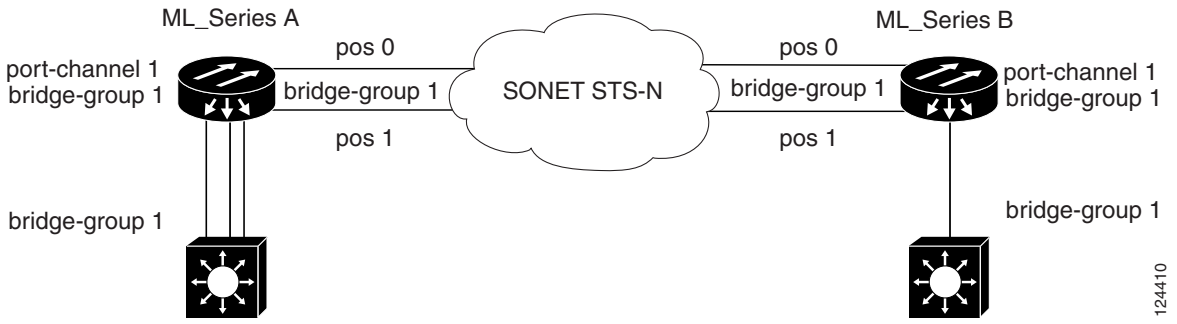
To assign POS interfaces to the POS channel, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface pos` *number*	Enters the interface configuration mode to configure the POS interface that you want to assign to the POS channel.
Step 2	`Router(config-if)# channel-group` *channel-number*	Assigns the POS interface to the POS channel. The channel number must be the same channel number that you assigned to the POS channel interface.
Step 3	`Router(config-if)# end`	Exits to privileged EXEC mode.
Step 4	`Router# copy running-config startup-config`	(Optional) Saves the configuration changes to NVRAM.

POS Channel Configuration Example

Figure 9-2 shows an example of POS channel configuration. The associated code for ML_Series A is provided in Example 9-3 and for ML_Series B in Example 9-4.

Figure 9-2 POS Channel Example



Example 9-3 ML_Series A Configuration

```
no ip routing
bridge 1 protocol ieee
!
!
interface Port-channel1
 no ip address
bridge-group 1
!
interface FastEthernet0
 no ip address
 bridge-group 1
```

```
!
interface POS0
channel-group 1
!
interface POS1
channel-group 1
```

Example 9-4 ML_Series B Configuration

```
bridge irb
bridge 1 protocol ieee
!
!
interface Port-channel1
bridge-group 1
!
interface FastEthernet0
bridge-group 1
!
interface POS0
channel-group 1
!
interface POS1
 no ip address
 channel-group 1
```

Understanding Encapsulation over FEC or POS Channel

When configuring encapsulation over FEC or POS, be sure to configure IEEE 802.1Q on the port-channel interface, not its member ports. However, certain attributes of port channel, such as duplex mode, need to be configured at the member port levels. Also make sure that you do not apply protocol-level configuration (such as an IP address or a bridge group assignment) to the member interfaces. All protocol-level configuration should be on the port channel or on its subinterface. You must configure IEEE 802.1Q encapsulation on the partner system of the EtherChannel as well.

Configuring Encapsulation over EtherChannel or POS Channel

To configure encapsulation over the FEC or POS channel, perform the following procedure, beginning in global configuration mode:

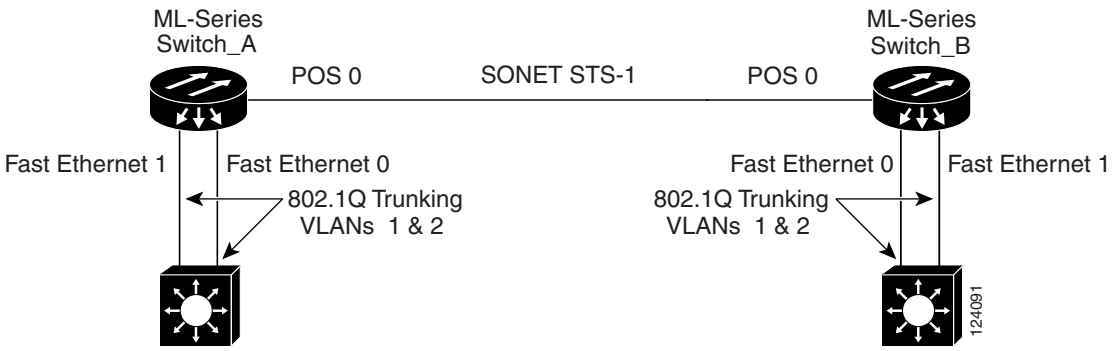
	Command	Purpose
Step 1	`Router(config)# interface port-channel` *channel-number.subinterface-number*	Configures the subinterface on the created port channel.
Step 2	`Router(config-subif)# encapsulation dot1q` *vlan-id*	Assigns the IEEE 802.1Q encapsulation to the subinterface.
Step 3	`Router(config-subif)# bridge-group` *bridge-group-number*	Assigns the subinterface to a bridge group.

	Command	Purpose
Step 4	`Router(config-subif)# end`	Exits to privileged EXEC mode. **Note** Optionally, you can remain in interface configuration mode and enable other supported interface commands to meet your requirements.
Step 5	`Router# copy running-config startup-config`	(Optional) Saves the configuration changes to NVRAM.

Encapsulation over EtherChannel Example

Figure 9-3 shows an example of encapsulation over EtherChannel. The associated code for ML_Series A is provided in Example 9-5 and for ML_Series B in Example 9-6.

Figure 9-3 Encapsulation over EtherChannel Example



This encapsulation over EtherChannel example shows how to set up two ONS 15310-CL nodes or ONS 15310-MA nodes with ML-Series cards to interoperate with two switches that also support IEEE 802.1Q encapsulation over EtherChannel. To set up this example, use the configurations in the following sections for both Switch A and Switch B.

Example 9-5 ML_Series A Configuration

```
hostname ML_Series_A
!
bridge irb
bridge 1 protocol ieee
bridge 2 protocol ieee
!
interface Port-channel1
hold-queue 150 in
!
interface Port-channel1.1
 encapsulation dot1Q 1 native
bridge-group 1
!
interface Port-channel1.2
 encapsulation dot1Q 2
bridge-group 2

!
```

```
interface FastEthernet0
channel-group 1
!
interface FastEthernet1
channel-group 1
!
interface POS0
!
interface POS0.1
 encapsulation dot1Q 1 native
 bridge-group 1
!
interface POS0.2
 encapsulation dot1Q 2
 bridge-group 2
```

Example 9-6 ML_Series B Configuration

```
hostname ML_Series_B
!
bridge irb
bridge 1 protocol ieee
bridge 2 protocol ieee
!
interface Port-channel1
hold-queue 150 in
!
interface Port-channel1.1
 encapsulation dot1Q 1 native
bridge-group 1
!
interface Port-channel1.2
 encapsulation dot1Q 2
bridge-group 2
!
interface FastEthernet0
channel-group 1
!
interface FastEthernet1
channel-group 1
!
interface POS0
!
interface POS0.1
 encapsulation dot1Q 1 native
 bridge-group 1
!
interface POS0.2
 encapsulation dot1Q 2
 bridge-group 2
!
```

Monitoring and Verifying EtherChannel and POS

After FEC or POS is configured, you can monitor its status using the **show interfaces port-channel** command.

Example 9-7 show interfaces port-channel Command

```
ML_Series# show int port-channel 9
Port-channel9 is down, line protocol is down
  Hardware is FEChannel, address is 0000.0000.0000 (bia 0000.0000.0000)
  Internet address is 192.26.24.22/25
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  ARP type: ARPA, ARP Timeout 04:00:00
    No. of active members in this channel: 0
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/300/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog, 0 multicast
     0 input packets with dribble condition detected
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
```

Load Balancing on the ML-Series cards

The load balancing for the Ethernet traffic on the portchannel is performed while sending the frame through a port channel interface based on the source MAC and destination MAC address of the Ethernet frame.

On a 2 port port channel interface, the Unicast Ethernet traffic (Learned MAC with unicast SA and DA) is transmitted on either first or second member of the port-channel based on the result of the "Exclusive OR" (XOR) operation applied on the second least significant bits (bit 1) of DA-MAC and SA-MAC. So, if the "XOR" result of the Ethernet frames SA-MAC second least significant bit and DA-MAC second least significant bit is 0 then the frame is sent on the first member and if the result is 1 then the frame is transmitted on the second member port of the port channel.

Table 9-1 MAC Based- 2- Port Channel Interface

Second Least Significant bit of the MAC-DA	Second Least Significant bit of the MAC-SA	XOR Result	Used Member Interface for the Frame Forwarding to the EtherChannel and/or Port Channel
0	0	0	Port 1
0	1	1	Port 2

Table 9-1 MAC Based- 2- Port Channel Interface

Second Least Significant bit of the MAC-DA	Second Least Significant bit of the MAC-SA	XOR Result	Used Member Interface for the Frame Forwarding to the EtherChannel and/or Port Channel
1	0	1	Port 2
1	1	0	Port 1

Table 9-2 IP Based- 2- Port Channel Interface

Second Least Significant bit of the IP-DA	Second Least Significant bit of the IP-SA	XOR Result	Used Member Interface for the Frame Forwarding to the EtherChannel and/or Port Channel
0	0	0	Port 1
0	1	1	Port 2
1	0	1	Port 2
1	1	0	Port 1

The Flood Ethernet traffic (Unknown MAC, Multicast and Broadcast frames) is transmitted on the first active member of the port-channel.

The routed IP Unicast traffic from the ML-Series towards the port channel ports is transmitted on either interface based on the result of the "Exclusive OR" (XOR) operation applied on the second least significant bits of the source and destination IP address of the IP packet. So if the "XOR" result of the IP packets Source Address least significant bit and Destination Address least significant bit is 0 then the frame is on the first member port and if the result is 1 then the frame is transmitted on the second member port.

On the 4 port EtherChannel the second and third least significant bits are used for load balancing.

Table 9-3 MAC Based - 4-Port Channel Interface

Third Least Significant bit of the MAC-DA	Third Least Significant bit of the MAC-SA	Second Least Significant bit of the MAC-DA	Second Least Significant bit of the MAC-SA	XOR Result	Used Member Interface for the Frame Forwarding to the EtherChannel and/or Port Channel
0	0	0	0	00	First
0	0	0	1	01	Second

Table 9-3 MAC Based - 4-Port Channel Interface

Third Least Significant bit of the MAC-DA	Third Least Significant bit of the MAC-SA	Second Least Significant bit of the MAC-DA	Second Least Significant bit of the MAC-SA	XOR Result	Used Member Interface for the Frame Forwarding to the EtherChannel and/or Port Channel
0	0	1	0	01	Second
0	0	1	1	00	First
0	1	0	0	10	Third
0	1	0	1	11	Fourth
0	1	1	0	11	Fourth
0	1	1	1	10	Third
1	0	0	0	10	Third
1	0	0	1	11	Fourth
1	0	1	0	11	Fourth
1	0	1	1	10	Third
1	1	0	0	00	First
1	1	0	1	01	Second
1	1	1	0	01	Second
1	1	1	1	00	First

Table 9-4 IP Based - 4-Port Channel Interface

Third Least Significant bit of the IP-DA	Third Least Significant bit of the IP-SA	Second Least Significant bit of the IP-DA	Second Least Significant bit of the IP-SA	XOR Result	Used Member Interface for the Frame Forwarding to the EtherChannel and/or Port Channel
0	0	0	0	00	First
0	0	0	1	01	Second
00	0	1	0	01	Second
0	0	1	1	00	First
0	1	0	0	10	Third
0	1	0	1	11	Fourth

Table 9-4 IP Based - 4-Port Channel Interface

Third Least Significant bit of the IP-DA	Third Least Significant bit of the IP-SA	Second Least Significant bit of the IP-DA	Second Least Significant bit of the IP-SA	XOR Result	Used Member Interface for the Frame Forwarding to the EtherChannel and/or Port Channel
0	1	1	0	11	Fourth
0	1	1	1	10	Second
1	0	0	0	10	Second
1	0	0	1	11	Third
1	0	1	0	11	Third
1	0	1	1	10	Second
1	1	0	0	00	First
1	1	0	1	01	Second
1	1	1	0	01	Second
1	1	1	1	00	First

The routed IP Multicast traffic from the ML-Series towards the RPR ring is transmitted on the first active member of the port channel.

CHAPTER 10

Configuring IRB on the ML-Series Card

This chapter describes how to configure integrated routing and bridging (IRB) for the ML-Series card. For more information about the Cisco IOS commands used in this chapter, refer to the *Cisco IOS Command Reference* publication.

This chapter includes the following major sections:

Caution Cisco Inter-Switch Link (ISL) and Cisco Dynamic Trunking Protocol (DTP) are not supported by the ML-Series, but the ML-Series broadcast forwards these formats. Using ISL or DTP on connecting devices is not recommended. Some Cisco devices attempt to use ISL or DTP by default.

Understanding Integrated Routing and Bridging

Your network might require you to bridge local traffic within several segments and have hosts on the bridged segments reach the hosts or ML-Series card on routed networks. For example, if you are migrating bridged topologies into routed topologies, you might want to start by connecting some of the bridged segments to the routed networks.

Using the integrated routing and bridging (IRB) feature, you can route a given protocol between routed interfaces and bridge groups within a single ML-Series card. Specifically, local or unroutable traffic is bridged among the bridged interfaces in the same bridge group, while routable traffic is routed to other routed interfaces or bridge groups.

Because bridging is in the data link layer and routing is in the network layer, they have different protocol configuration models. With IP, for example, bridge group interfaces belong to the same network and have a collective IP network address. In contrast, each routed interface represents a distinct network and has its own IP network address. It uses the concept of a Bridge Group Virtual Interface (BVI) to enable these interfaces to exchange packets for a given protocol.

A BVI is a virtual interface within the ML-Series card that acts like a normal routed interface. A BVI does not support bridging but actually represents the corresponding bridge group to routed interfaces within the ML-Series card. It also gives the user an IP management interface for the bridge group. The interface number is the link between the BVI and the bridge group.

Before configuring IRB, consider the following:

- The default routing/bridging behavior in a bridge group (when IRB is enabled) is to bridge all packets. Make sure that you explicitly configure routing on the BVI for IP traffic.
- Packets of unroutable protocols such as local-area transport (LAT) are always bridged. You cannot disable bridging for the unroutable traffic.
- Protocol attributes should not be configured on the bridged interfaces when you are using IRB to bridge and route a given protocol. You can configure protocol attributes on the BVI, but you cannot configure bridging attributes on the BVI.
- A bridge links several network segments into one large, flat network. To bridge a packet coming from a routed interface among bridged interfaces, the bridge group should be represented by one interface.
- All ports in a BVI group must have matching maximum transmission unit (MTU) settings.

Configuring IRB

The process of configuring integrated routing and bridging consists of the following tasks:

1. Configure bridge groups and routed interfaces.
   a. Enable bridging.
   b. Assign interfaces to the bridge groups.
   c. Configure the routing.
2. Enable IRB.
3. Configure the BVI.
   a. Enable the BVI to accept routed packets.
   b. Enable routing on the BVI.
4. Configure IP addresses on the routed interfaces.
5. Verify the IRB configuration.

When you configure the BVI and enable routing on it, packets that come in on a routed interface destined for a host on a segment that is in a bridge group are routed to the BVI and forwarded to the bridging engine. From the bridging engine, the packet exits through a bridged interface. Similarly, packets that come in on a bridged interface but are destined for a host on a routed interface go first to the BVI. The BVI forwards the packets to the routing engine that sends them out on the routed interface.

To configure a bridge group and an interface in the bridge group, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	`ML_Series(config)# bridge` *bridge-group* `protocol {ieee | rstp}`	Defines one or more bridge groups.
Step 2	`ML_Series(config)# interface` *type number*	Enters interface configuration mode.
Step 3	`ML_Series(config-if)# bridge-group` *bridge-group*	Assigns the interface to the specified bridge group.

	Command	Purpose
Step 4	`ML_Series(config-if)# ip address ip-address ip-address-subnet-mask`	Configures IP addresses on routed interfaces.
Step 5	`ML_Series(config-if)# end`	Returns to privileged EXEC mode.

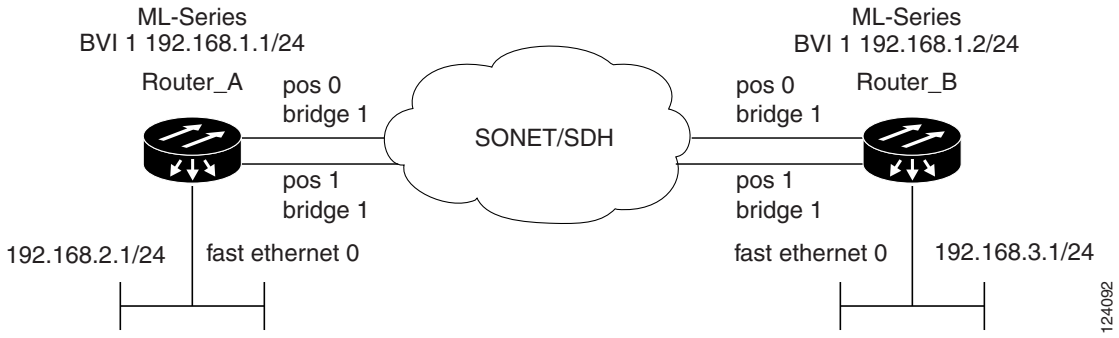
To enable and configure IRB and BVI, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	`ML_Series(config)# bridge irb`	Enables IRB. Allows bridging of traffic.
Step 2	`ML_Series(config)# interface bvi bridge-group`	Configures the BVI by assigning the number of the corresponding bridge group to the BVI. Each bridge group can have only one corresponding BVI.
Step 3	`ML_Series(config-if)# ip address ip-address ip-address-subnet-mask`	Configures IP addresses on routed interfaces.
Step 4	`ML_Series(config-if)# exit`	Exits the interface configuration mode.
Step 5	`ML_Series(config)# bridge bridge-group route protocol`	Enables a BVI to accept and route routable packets received from its corresponding bridge group. Enter this command for each protocol that you want the BVI to route from its corresponding bridge group to other routed interfaces.
Step 6	`ML_Series(config)# end`	Returns to the privileged EXEC mode.
Step 7	`ML_Series# copy running-config startup-config`	(Optional) Saves your configuration changes to NVRAM.

IRB Configuration Example

Figure 10-1 shows an example of IRB configuration. Example 10-1 shows the configuration code for ML_Series A, and Example 10-2 shows the configuration code for ML_Series B.

Figure 10-1 Configuring IRB



Example 10-1 Configuring ML_Series A

```
bridge irb
bridge 1 protocol ieee
 bridge 1 route ip
!
!
interface FastEthernet0
 ip address 192.168.2.1 255.255.255.0

!
interface POS0
no ip address
bridge-group 1
!
interface POS1
no ip address
bridge-group 1
!
interface BVI1
 ip address 192.168.1.1 255.255.255.0
!
router ospf 1
 log-adjacency-changes
 network 192.168.1.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0
```

Example 10-2 Configuring ML_Series B

```
bridge irb
bridge 1 protocol ieee
 bridge 1 route ip
!
!
interface FastEthernet0
 ip address 192.168.3.1 255.255.255.0

!
interface POS0
 no ip address
bridge-group 1
!
interface POS1
no ip address
bridge-group 1
!
interface BVI1
 ip address 192.168.1.2 255.255.255.0
!
router ospf 1
 log-adjacency-changes
 network 192.168.1.0 0.0.0.255 area 0
 network 192.168.3.0 0.0.0.255 area 0
```

Monitoring and Verifying IRB

Table 10-1 shows the privileged EXEC commands for monitoring and verifying IRB.

Table 10-1 Commands for Monitoring and Verifying IRB

Command	Purpose
`Router# show interfaces bvi bvi-interface-number`	Shows BVI information, such as the BVI MAC address and processing statistics. The *bvi-interface-number* is the number of the bridge group assigned to the BVI interface.
`Router# show interfaces [type-number] irb`	Shows BVI information for the following: • Protocols that this bridged interface can route to the other routed interface (if this packet is routable). • Protocols that this bridged interface bridges

The following is sample output from the **show interfaces bvi** (Example 10-3) and **show interfaces irb** commands (Example 10-4):

Example 10-3 show interfaces bvi

```
Router# show interfaces bvi 22
BVI22 is down, line protocol is down
  Hardware is BVI, address is 0012.0101.362c (bia 0000.0000.0000)
  Internet address is 192.192.192.194/24
  MTU 1500 bytes, BW 100000 Kbit, DLY 5000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
 Encapsulation: ARPA, loopback not set
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
```

Example 10-4 show interfaces irb

```
Router# show interfaces irb
BVI 22
pkts_error_giants         0

  Hash Len    Address      Matches  Act      Type
pkts_error_runts
  0x00:  0 ffff.ffff.ffff         0 RCV Physical broadcast
pkts_mcast                00 no
  0x2B:  0 0012.0101.362a         0 RCV Interface MAC addressfailures, 0 out
align_errors              0
  0xC0:  0 0100.0ccc.cccc         0 RCV CDPcarrier transitions
Overruns
 Bridged protocols on POS0:
```

```
  clns       ip
Software MAC address filter on POS0
  Hash Len    Address      Matches  Act      Type
  0x00:  0 ffff.ffff.ffff         0 RCV Physical broadcast
  0x25:  0 0012.0101.3624         0 RCV Interface MAC address
  0x29:  0 0012.0101.3628         0 RCV Interface MAC address
  0xC0:  0 0100.0ccc.cccc         0 RCV CDP
  0xC2:  0 0180.c200.0000         0 RCV IEEE spanning tree
POS1
Bridged protocols on POS1:
  clns       ip
Software MAC address filter on POS1
  Hash Len    Address      Matches  Act      Type
  0x00:  0 ffff.ffff.ffff         0 RCV Physical broadcast
  0x24:  0 0012.0101.3625         0 RCV Interface MAC address
  0x29:  0 0012.0101.3628         0 RCV Interface MAC address
  0xC0:  0 0100.0ccc.cccc         0 RCV CDP
  0xC2:  0 0180.c200.0000         0 RCV IEEE spanning tree
```

Table 10-2 describes significant fields shown in the display.

Table 10-2 show interfaces irb Field Descriptions

Field	Description
`Routed protocols on…`	List of the routed protocols configured for the specified interface.
`Bridged protocols on…`	List of the bridged protocols configured for the specified interface.
`Software MAC address filter on…`	Table of software MAC address filter information for the specified interface.
`Hash`	Hash key/relative position in the keyed list for this MAC-address entry.
`Len`	Length of this entry to the beginning element of this hash chain.
`Address`	Canonical (Ethernet ordered) MAC address.
`Matches`	Number of received packets matched to this MAC address.
`Routed protocols on…`	List of the routed protocols configured for the specified interface.
`Bridged protocols on…`	List of the bridged protocols configured for the specified interface.

CHAPTER 11

Configuring Quality of Service on the ML-Series Card

This chapter describes the Quality of Service (QoS) features built into your ML-Series card. It also describes how to map QoS scheduling at both the system and interface levels.

This chapter contains the following major sections:

The ML-Series card employs the Cisco IOS Modular QoS Command-line Interface (MQC). For more information on general MQC configuration, refer to the following Cisco IOS documents:

- *Cisco IOS Quality of Service Solutions Configuration Guide, Release 12.2* at this URL: http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122mindx/l22index.htm
- *Cisco IOS Quality of Service Solutions Command Reference, Release 12.2* at this URL: http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fqos_r/index.htm

Understanding QoS

The ML-Series card multiplexes multiple IP/Ethernet services onto the SONET circuit and dynamically allocates transmission bandwidth to data services based on data service requirements. This allows the network to operate at a significantly higher level of utilization. To support service-level agreements (SLAs), this dynamic allocation must accommodate the service elements of bandwidth, including loss and delay. The characteristics of these service elements make up QoS.

The QoS mechanism has three basic steps. It classifies types of traffic, specifies what action to take against a type of traffic, and specifies where the action should take place.

Priority Mechanism in IP and Ethernet

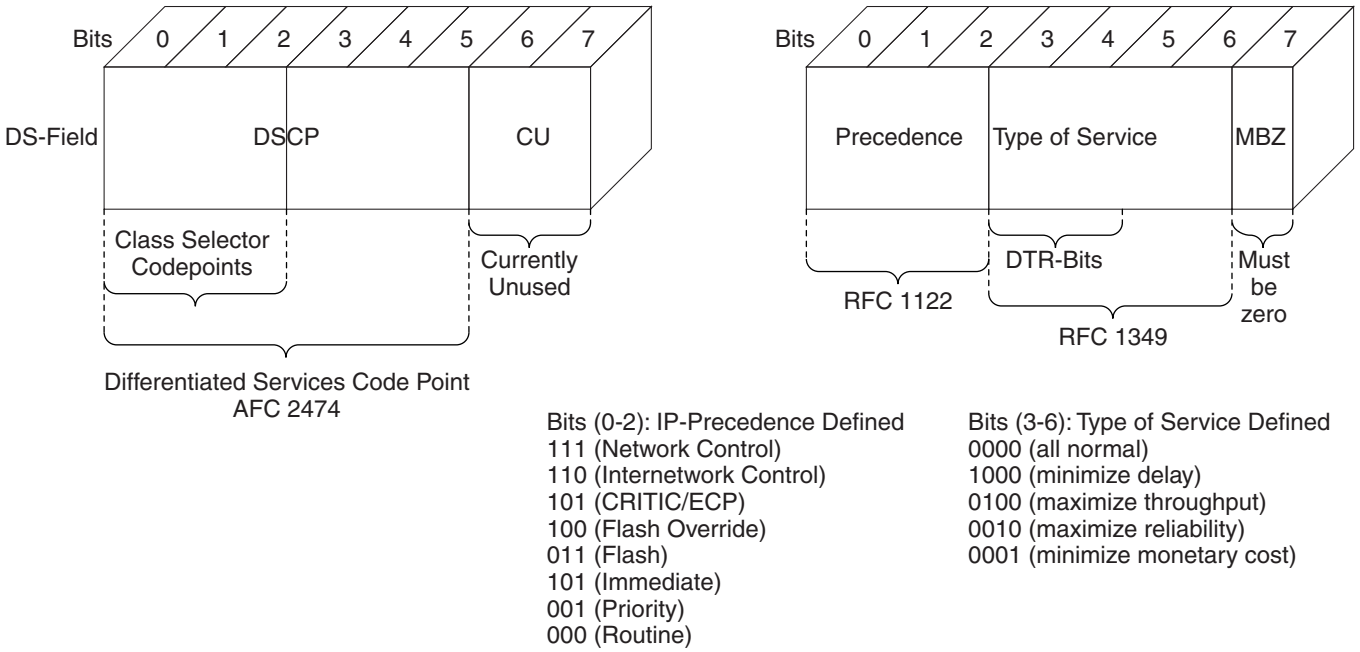
For any QoS service to be applied to data, there must be a way to mark or identify an IP packet or an Ethernet frame. When identified, a specific priority can be assigned to each individual IP packet or Ethernet frame. The IP Precedence or the IP Differentiated Services Code Point (DSCP) field prioritizes the IP packets, and the Ethernet class of service (IEEE 802.1p defined class of service [CoS]) is used for the Ethernet frames. IP precedence and Ethernet CoS are further described in the following sections.

IP Precedence and Differentiated Services Code Point

IP precedence uses the three precedence bits in the IPv4 header's ToS (type of service) field to specify class of service for each IP packet (RFC 1122). The most significant three bits of the IPv4 ToS field provide up to eight distinct classes, of which six are used for classifying services and the remaining two are reserved. On the edge of the network, the IP precedence is assigned by the client device or the ML Series, so that each subsequent network element can provide services based on the determined policy or the SLA.

IP DSCP uses the six bits in the IPv4 header to specify class of service for each IP packet (RFC 2474). Figure 11-1 illustrates IP precedence and DSCP. The DSCP field classifies packets into any of the 64 possible classes. On the network edge, the IP DSCP is assigned by the client device or the ML Series, so that each subsequent network element can provide services based on the determined policy or the SLA.

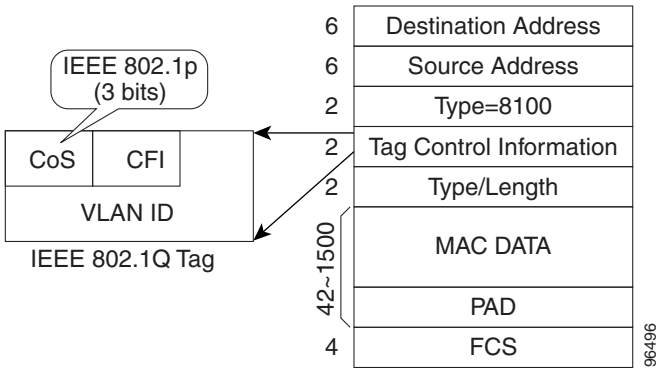
Figure 11-1 IP Precedence and DSCP



Ethernet CoS

Ethernet CoS refers to three bits within a four byte IEEE 802.1Q (VLAN) header used to indicate the priority of the Ethernet frame as it passes through a switched network. The CoS bits in the IEEE 802.1Q header are commonly referred to as the IEEE 802.1p bits. There are three CoS bits that provide eight classes, matching the number delivered by IP precedence. In many real-world networks, a packet might traverse both Layer 2 and Layer 3 domains. To maintain QoS across the network, the IP ToS can be mapped to the Ethernet CoS and vice versa, for example, in linear or one-to-one mapping, because each mechanism supports eight classes. Similarly, a set of DSCP values (64 classes) can be mapped into each of the eight individual Ethernet CoS values. Figure 11-2 is an IEEE 802.1Q Ethernet frame, which consists of a 2-byte Ethertype and a 2-byte tag (IEEE 802.1Q Tag) on the Ethernet protocol header.

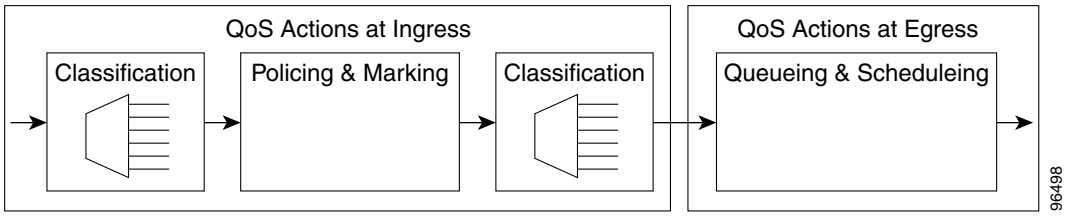
Figure 11-2 Ethernet Frame and the CoS Bit (IEEE 802.1p)



ML-Series QoS

The ML-Series QoS classifies each packet in the network based on its input interface, bridge group (VLAN), Ethernet CoS, IP precedence, IP DSCP, or resilient packet ring (RPR)-CoS. After they are classified into class flows, further QoS functions can be applied to each packet as it traverses the card. Figure 11-3 illustrates the ML-Series QoS flow.

Figure 11-3 ML-Series QoS Flow



Policing provided by the ML-Series card ensures that attached equipment does not submit more than a predefined amount of bandwidth (Rate Limiting) into the network. The policing feature can be used to enforce the committed information rate (CIR) and the peak information rate (PIR) available to a customer at an interface. Policing also helps characterize the statistical nature of the information allowed into the network so that traffic engineering can more effectively ensure that the amount of committed bandwidth is available on the network, and the peak bandwidth is over-subscribed with an appropriate ratio. The policing action is applied per classification.

Priority marking can set the Ethernet IEEE 802.1p CoS bits or RPR-CoS bits as they exit the ML-Series card. The marking feature operates on the outer IEEE 802.1p tag, and provides a mechanism for tagging packets at the ingress of a QinQ packet. The subsequent network elements can provide QoS based only on this service-provider-created QoS indicator.

Per-class flow queuing enables fair access to excess network bandwidth, allows allocation of bandwidth to support SLAs, and ensures that applications with high network resource requirements are adequately served. Buffers are allocated to queues dynamically from a shared resource pool. The allocation process incorporates the instantaneous system load as well as the allocated bandwidth to each queue to optimize buffer allocation. Congestion management on the ML-Series is performed through a tail drop mechanism along with discard eligibility on the egress scheduler.

The ML-Series uses a Weighted Deficit Round Robin (WDRR) scheduling process to provide fair access to excess bandwidth as well as guaranteed throughput to each class flow.

Admission control is a process that is invoked each time that service is configured on the ML-Series card to ensure that QoS resources are not overcommitted. In particular, admission control ensures that no configurations are accepted when the sum of committed bandwidths on an interface exceeds the total bandwidth on the interface.

Classification

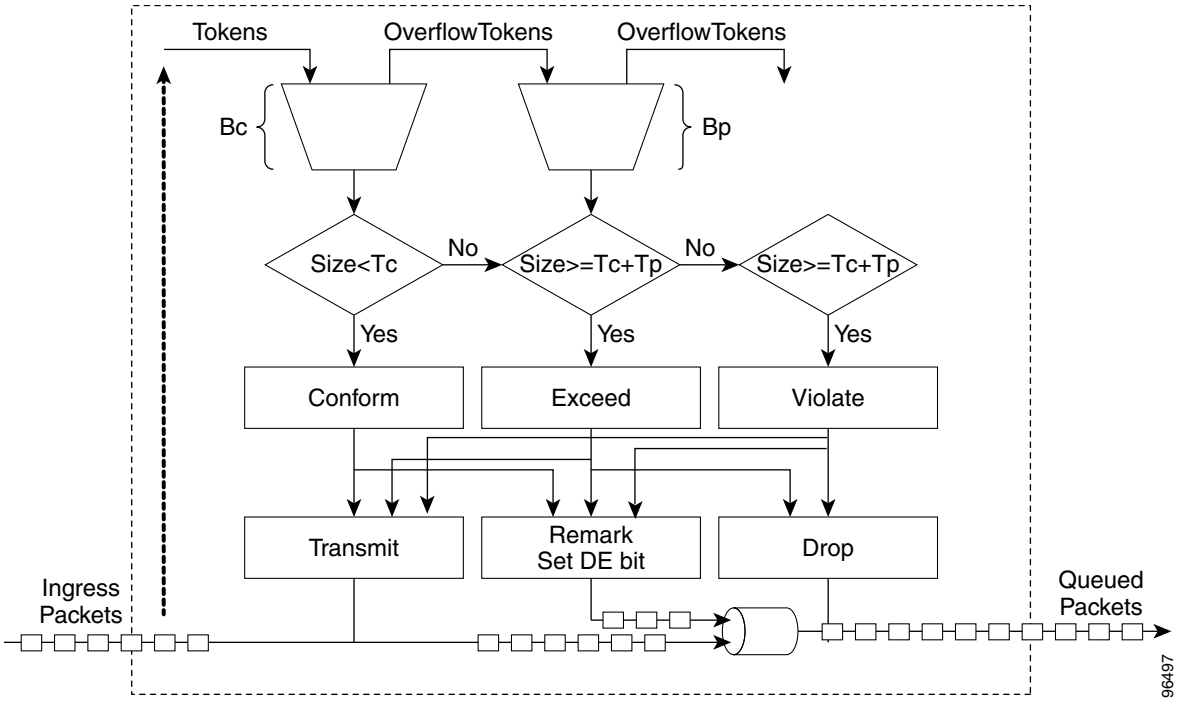
Classification can be based on any single packet classification criteria or a combination (logical AND and OR). Classification of packets is configured using the Modular CLI **class-map** command. For traffic transiting the RPR, only the input interface and/or the RPR-CoS can be used as classification criteria.

Policing

Dual leaky bucket policer is a process where the first bucket (CIR bucket) is filled with tokens at a known rate (CIR), which is a parameter that can be configured by the operator. Figure 11-4 illustrates the dual leaky bucket policer model. The tokens fill the bucket up to a maximum level, which is the amount of burstable committed (BC) traffic on the policer. The nonconforming packets of the first bucket are the overflow packets, which are passed to the second leaky bucket (the PIR bucket). The second leaky bucket is filled with these tokens at a known rate (PIR), which is a parameter that can be configured by the operator. The tokens fill the PIR bucket up to a maximum level (BP), which is the amount of peak burstable traffic on the policer. The nonconform packets of the second bucket are the overflow packets, which can be dropped or marked according to the policer definition.

On the dual leaky bucket policer, the packets conforming to the CIR are conform packets, the packets not conforming to CIR but conforming to PIR are exceed packets, and the packets not conforming to either the PIR or CIR are violate packets.

Figure 11-4 Dual Leaky Bucket Policer Model



Marking and Discarding with a Policer

On the ML-Series card's policer, the conform packets can be transmitted or marked and transmitted. The exceed packets can be transmitted, marked and transmitted, or dropped. The violating packets can be transmitted, marked and transmitted, or dropped. The primary application of the dual-rate or three-color policer is to mark the conform packets with CoS bit 2l, mark the exceed packet with CoS bit 1, and discard the violated packets so all the subsequent network devices can implement the proper QoS treatment per frame/packet basis based on these priority marking without knowledge of each SLA.

In some cases, it might be desirable to discard all traffic of a specific ingress class. This can be accomplished by using a police command of the following form with the class: **police 96000 conform-action drop exceed-action drop**.

If a marked packet has a provider-supplied Q-tag inserted before transmission, the marking only affects the provider Q-tag. If a Q-tag is received, it is re-marked. If a marked packet is transported over the RPR ring, the marking also affects the RPR-CoS bit.

If a Q-tag is inserted (QinQ), the marking affects the added Q-tag. If the ingress packet contains a Q-tag and is transparently switched, the existing Q-tag is marked. In case of a packet without any Q-tag, the marking does not have any significance.

The local scheduler treats all nonconforming packets as discard eligible regardless of their CoS setting or the global cos commit definition. For RPR implementation, the discard eligible (DE) packets are marked using the DE bit on the RPR header. The discard eligibility based on the CoS commit or the policing action is local to the ML-Series card scheduler, but it is global for the RPR ring.

Queuing

ML-Series card queuing uses a shared buffer pool to allocate memory dynamically to different traffic queues. The ML-100T-8 has 1.5 MB of packet buffer memory.

Each queue has an upper limit on the allocated number of buffers based on the class bandwidth assignment of the queue and the number of queues configured. This upper limit is typically 30 percent to 50 percent of the shared buffer capacity. Dynamic buffer allocation to each queue can be reduced based on the number of queues needing extra buffering. The dynamic allocation mechanism provides fairness in proportion to service commitments as well as optimization of system throughput over a range of system traffic loads.

The Low Latency Queue (LLQ) is defined by setting the weight to infinity or committing 100 percent bandwidth. When a LLQ is defined, a policer should also be defined on the ingress for that specific class to limit the maximum bandwidth consumed by the LLQ; otherwise there is a potential risk of LLQ occupying the whole bandwidth and starving the other unicast queues.

The ML-Series includes support for 400 user-definable queues, which are assigned per the classification and bandwidth allocation definition. The classification used for scheduling classifies the frames/packet after the policing action, so if the policer is used to mark or change the CoS bits of the ingress frames/packet, the new values are applicable for the classification of traffic for queuing and scheduling. The ML-Series provides buffering for 4000 packets.

Scheduling

Scheduling is provided by a series of schedulers that perform a WDRR as well as priority scheduling mechanisms from the queued traffic associated with each egress port.

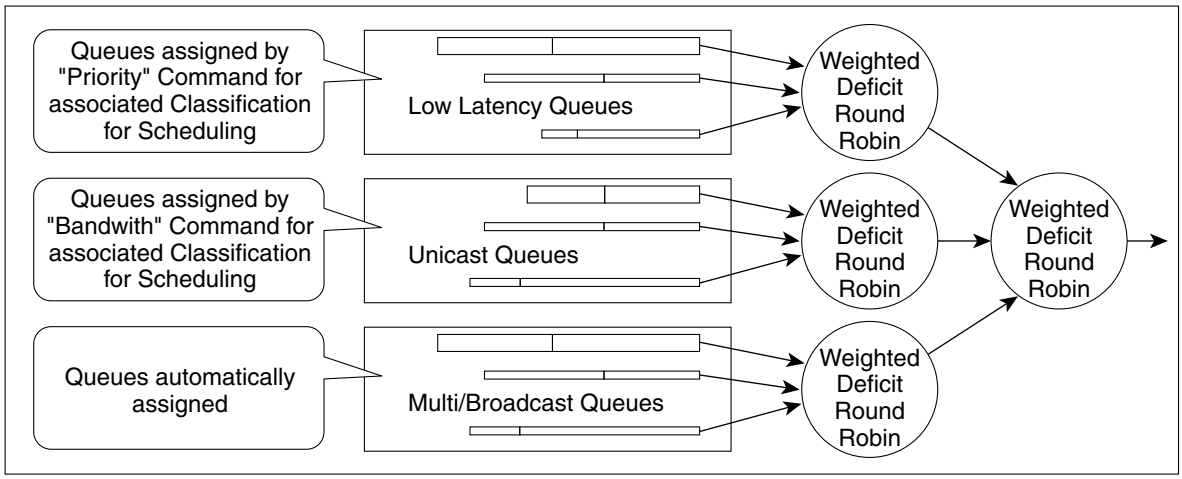
Though ordinary round robin servicing of queues can be done in constant time, unfairness occurs when different queues use different packet sizes. Deficit Round Robin (DRR) scheduling solves this problem. If a queue was not able to send a packet in its previous round because its packet size was too large, the remainder from the previous amount of credits that the queue got in each previous round (quantum) is added to the quantum for the next round.

WDRR extends the quantum idea from the DRR to provide weighted throughput for each queue. Different queues have different weights, and the quantum assigned to each queue in its round is proportional to the relative weight of the queue among all the queues serviced by that scheduler.

Weights are assigned to each queue as a result of the service provisioning process. When coupled with policing and policy mapping provisioning, these weights and the WDRR scheduling process ensure that QoS commitments are provided to each service flow.

Figure 11-5 illustrates the ML-Series card's queuing and scheduling.

Figure 11-5 Queuing and Scheduling Model



The weighting structure allows traffic to be scheduled at 1/2048 of the port rate. This equates to approximately 49 kbps for traffic exiting a FastEthernet port.

The unicast queues are created as the output service policy implementation on the egress ports. Each unicast queue is assigned with a committed bandwidth and the weight of the queue is determined by the normalization of committed bandwidth of all defined unicast queues for that port. The traffic beyond the committed bandwidth on any queue is treated by the scheduler according to the relative weight of the queue.

The LLQ is created as the output service policy implementation on the egress ports. Each LLQ is assigned with a committed bandwidth of 100 percent and is served with lower latency. To limit the bandwidth usage by the LLQ, a strict policer needs to be implemented on the ingress for the LLQ traffic classes.

The DE allows some packets to be treated as committed and some as discard-eligible on the scheduler. For the Ethernet frames, the CoS (IEEE 802.1p) bits are used to identify committed and discard eligible packets, where the RPR-CoS and the DE bits are used for RPR traffic. When congestion occurs and a queue begins to fill, the DE packets hit a lower tail-drop threshold than the committed packets. Committed packets are not dropped until the total committed load exceeds the interface output. The tail-drop thresholds adjust dynamically in the card to maximize use of the shared buffer pool while guaranteeing fairness under all conditions.

Control Packets and L2 Tunneled Protocols

The control packets originated by the ML-Series card have a higher priority than data packets. The external Layer 2 and Layer 3 control packets are handled as data packets and assigned to broadcast queues. Bridge protocol data unit (BPDU) prioritization in the ML-Series card gives Layer 2-tunneled BPDU sent out the multicast/broadcast queue a higher discard value and therefore a higher priority than than other packets in the multicast/broadcast queue. The Ethernet CoS (IEEE 802.1p) for Layer 2-tunneled protocols can be assigned by the ML-Series card.

Egress Priority Marking

Egress priority marking allows the operator to assign the IEEE 802.1p CoS bits of packets that exit the card. This marking allows the operator to use the CoS bits as a mechanism for signaling to downstream nodes the QoS treatment that the packet should be given. This feature operates on the outer-most IEEE 802.1p CoS field. When used with the QinQ feature, priority marking allows the user traffic (inner Q-tag) to traverse the network transparently, while providing a means for the network to internally signal QoS treatment at Layer 2.

Priority marking follows the classification process, and therefore any of the classification criteria identified earlier can be used as the basis to set the outgoing IEEE 802.1p CoS field. For example, a specific CoS value can be mapped to a specific bridge group.

Priority marking is configured using the MQC **set-cos** command. If packets would otherwise leave the card without an IEEE 802.1Q tag, then the **set-cos** command has no effect on that packet. If an IEEE 802.1Q tag is inserted in the packet (either a normal tag or a QinQ tag), the inserted tag has the set-cos priority. If an IEEE 802.1Q tag is present on packet ingress and retained on packet egress, the priority of that tag is modified. If the ingress interface is an QinQ access port and the **set-cos** policy-map classifies based on ingress tag priority, this classifies based on the user priority. This is a way to allow the user-tag priority to determine the SP tag priority. When a packet does not match any **set-cos** policy-map, the priority of any preserved tag is unchanged and the priority of any inserted IEEE 802.1Q tag is set to 0.

The **set-cos** command on the output service policy is only applied to unicast traffic. Priority marking for multicast/broadcast traffic can only be achieved by the **set-cos** action of the policing process on the input service policy.

Ingress Priority Marking

Ingress priority marking can be done for all input packets of a port, for all input packets matching a classification, or based on a measured rate. Marking of all packets of an input class can also be done with a policing command of the form **police 96000 conform-action set-cos-transmit exceed-action set-cos-transmit**. Using this command with a policy map that contains only the “class-default” will mark all ingress packets to the value. Rate based priority marking is discussed in the “Marking and Discarding with a Policer” section on page 11-5.

QinQ Implementation

The hierarchical VLAN or IEEE 802.1Q tunneling feature enables the service provider to transparently carry the customer VLANs coming from any specific port (UNI) and transport them over the service provider network. This feature is also known as QinQ, which is performed by adding an additional IEEE 802.1Q tag on every customer frame.

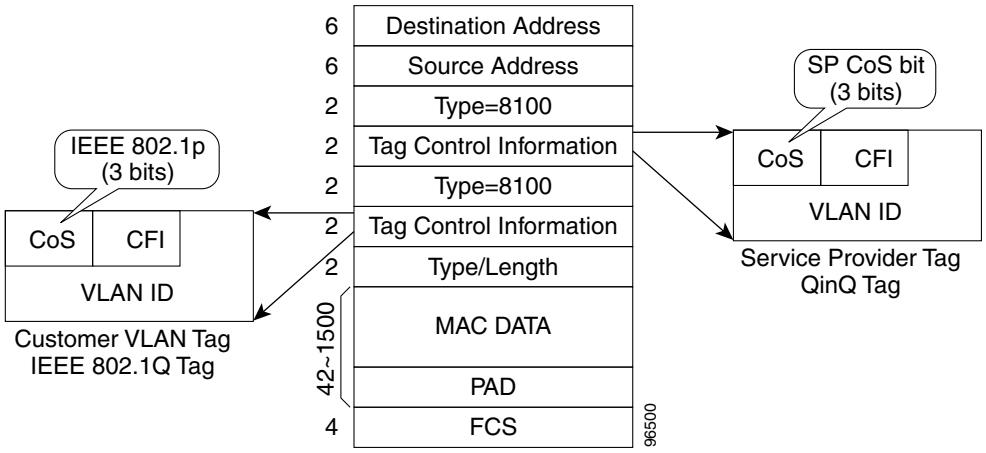
Using the QinQ feature, service providers can use a single VLAN to support customers with multiple VLANs. QinQ preserves customer VLAN IDs and segregates traffic from different customers within the service-provider infrastructure, even when traffic from different customers originally shared the same VLAN ID. The QinQ also expands VLAN space by using a VLAN-in-VLAN hierarchy and tagging the tagged packets. When the service provider (SP) tag is added, the QinQ network typically loses any visibility to the IP header or the customer Ethernet IEEE 802.1Q tag on the QinQ encapsulated frames.

On the ML-Series cards, the QinQ access ports (IEEE 802.1Q tunnel ports or QinQ UNI ports) have visibility to the customer CoS and the IP precedence or IP DSCP values; therefore, the SP tag can be assigned with proper CoS bit, which would reflect the customer IP precedence, IP DSCP, or CoS bits. In

the QinQ network, the QoS is then implemented based on the IEEE 802.1p bit of the SP tag. The ML-Series cards do not have visibility into the customer CoS, IP precedence, or DSCP values after the packet is double-tagged (because it is beyond the entry point of the QinQ service).

Figure 11-6 illustrates the QinQ implementation on the ML-Series card.

Figure 11-6 QinQ Implementation on the ML-Series Card



The ML-Series cards can be used as the IEEE 802.1Q tunneling device for the QinQ network and also provide the option to copy the customer frame's CoS bit into the CoS bit of the added QinQ tag. This allows the service provider QinQ network to be fully aware of the necessary QoS treatment for each individual customer frame.

Flow Control Pause and QoS

When flow control and policy-map are both enabled for an interface and the policy-map is configured only with 'class-default' having policer action, flow control handles the bandwidth. If the policer gets noncompliant flow, then it drops or demarks the packets using the policer definition of the interface.

Note QoS and policing are not supported on the ML-Series card interface when link aggregation is used.

Note Egress shaping is not supported on the ML-Series cards.

QoS on RPR

For VLAN bridging over RPR, all ML-Series cards on the ring must be configured with the base RPR and RPR QoS configuration. SLA and bridging configurations are only needed at customer RPR access points, where IEEE 802.1Q VLAN CoS is copied to the RPR CoS. This IEEE 802.1Q VLAN CoS copying can be overwritten with a **set-cos** *action* command. The CoS commit rule applies at the RPR ring ingress. Transit RPR ring traffic is classified on CoS only.

If the packet does not have a VLAN header, the RPR CoS for non-VLAN traffic is set using the following rules:

1. The default CoS is 0.

2. If the packet comes in with an assigned CoS, the assigned CoS replaces the default. If an IP packet originates locally, the IP precedence setting replaces the CoS setting.
3. The input policy map has a **set-cos** action.
4. The output policy map has a **set-cos** action (except for broadcast or multicast packets).

The RPR header contains a CoS value and DE indicator. The RPR DE is set for noncommitted traffic.

Configuring QoS

This section describes the tasks for configuring the ML-Series card QoS functions using the MQC. The ML-Series card does not support the full set of MQC functionality.

To configure and enable class-based QoS features, perform the procedures described in the following sections:

- Creating a Traffic Class, page 11-10
- Creating a Traffic Policy, page 11-11
- Attaching a Traffic Policy to an Interface, page 11-15
- Configuring CoS-Based QoS, page 11-16

For QoS configuration examples, see the "QoS Configuration Examples" section on page 11-17.

Creating a Traffic Class

The **class-map** global configuration command is used to create a traffic class. The syntax of the **class-map** command is as follows:

class-map [**match-any** | **match-all**] *class-map-name*
no class-map [**match-any** | **match-all**] *class-map-name*

The match-all and match-any options need to be specified only if more than one match criterion is configured in the traffic class. The **class-map match-all** command is used when all of the match criteria in the traffic class must be met for a packet to match the specified traffic class. The **class-map match-any** command is used when only one of the match criterion in the traffic class must be met for a packet to match the specified traffic class. If neither the **match-all** nor **match-any** keyword is specified, the traffic class behaves in a manner consistent with **class-map match-all** command.

To create a traffic class containing match criteria, use the **class-map** global configuration command to specify the traffic class name, and then use the **match** commands in Table 11-1, as needed.

Table 11-1 Traffic Class Commands

Command	Purpose
`ML_Series(config)# class-map class-map-name`	Specifies the user-defined name of the traffic class. Names can be a maximum of 40 alphanumeric characters. If **match-all** or **match-any** is not specified, traffic must match all the match criteria to be classified as part of the traffic class. There is no default-match criteria. Multiple match criteria are supported. The command matches either all or any of the criteria, as controlled by the **match-all** and **match-any** subcommands of the **class-map** command. **Note** The ML-100T-8 supports a maximum of 126 user-defined class maps, plus one default class map named "class-default". The ML-Series card on the ONS 15454 SONET/SDH supports a maximum of 254 user-defined class maps, plus one default class map named "class-default".
`ML_Series(config)# class-map match-all class-map-name`	Specifies that all match criteria must be met for traffic entering the traffic class to be classified as part of the traffic class.
`ML_Series(config)# class-map match-any class-map-name`	Specifies that one of the match criteria must be met for traffic entering the traffic class to be classified as part of the traffic class.
`ML_Series(config-cmap)# match any`	Specifies that all packets will be matched.
`ML_Series(config-cmap)# match bridge-group bridge-group-number`	Specifies the bridge-group-number against whose contents packets are checked to determine if they belong to the class.
`ML_Series(config-cmap)# match cos cos-number`	Specifies the CoS value against whose contents packets are checked to determine if they belong to the class.
`ML_Series(config-cmap)# match input-interface interface-name`	Specifies the name of the input interface used as a match criterion against which packets are checked to determine if they belong to the class. The shared packet ring (SPR) interface used in RPR (SPR1) is a valid interface-name for the ML-Series card. For more information on the SPR interface, see Chapter 14, "Configuring Resilient Packet Ring on the ML-Series Card." The **input-interface** choice is not valid when applied to the INPUT of an interface (redundant).
`ML_Series(config-cmap)# match ip dscp ip-dscp-value`	Specifies up to eight DSCP values used as match criteria. The value of each service code point is from 0 to 63.
`ML_Series (config-cmap)# match ip precedence ip-precedence-value`	Specifies up to eight IP precedence values used as match criteria.

Creating a Traffic Policy

To configure a traffic policy, use the **policy-map** global configuration command to specify the traffic policy name, and use the following configuration commands to associate a traffic class, which was configured with the **class-map** command and one or more QoS features. The traffic class is associated

with the traffic policy when the **class** command is used. The **class** command must be issued after entering policy-map configuration mode. After entering the **class** command, you are automatically in policy-map class configuration mode, which is where the QoS policies for the traffic policy are defined.

When the bandwidth or priority action is used on any class in a policy map, then there must be a class, defined by the **match-any** command, that has a bandwidth or priority action in that policy map. This is to ensure that all traffic can be classified into a default class that has some assigned bandwidth. A minimum bandwidth can be assigned if the class is not expected to be used or if no reserved bandwidth is desired for default traffic.

The QoS policies that can be applied in the traffic policy in policy-map class configuration mode are detailed in the following example.

The syntax of the **policy-map** command is:

policy-map *policy-name*
no policy-map *policy-name*

The syntax of the **class** command is:

class *class-map-name*
no class *class-map-name*

All traffic that fails to meet the matching criteria belongs to the default traffic class. The default traffic class can be configured by the user, but cannot be deleted.

To create a traffic policy, use the commands in Table 11-2 as needed.

Table 11-2 Traffic Policy Commands

Command	Purpose
`ML_Series (config)# policy-map policy-name`	Specifies the name of the traffic policy to configure. Names can be a maximum of 40 alphanumeric characters.
`ML_Series (config-pmap)# class class-map-name`	Specifies the name of a predefined traffic class, which was configured with the **class-map** command, used to classify traffic to the traffic policy.
`ML_Series (config-pmap)# class class-default`	Specifies the default class to be created as part of the traffic policy.

Table 11-2 Traffic Policy Commands (continued)

Command	Purpose
`ML_Series (config-pmap-c)# bandwidth {bandwidth-kbps | percent percent}`	Specifies a minimum bandwidth guarantee to a traffic class in periods of congestion. A minimum bandwidth guarantee can be specified in kbps or by a percentage of the overall available bandwidth. Valid choices for the ML-Series cards are: • Rate in kilobits per second • Percent of total available bandwidth (1 to 100) If multiple classes and bandwidth actions are specified in a single policy map, they must use the same choice in specifying bandwidth (kilobits or percent). **Note** When using the **bandwidth** command, excess traffic (beyond the configured commit) is allocated any available bandwidth in proportion to the relative bandwidth commitment of its traffic class compared to other traffic classes. Excess traffic from two classes with equal commits has equal access to available bandwidth. Excess traffic from a class with a minimum commit might receive only a minimum share of available bandwidth compared to excess bandwidth from a class with a high commit. **Note** The true configurable bandwidth in kilobits per second is per port and depends on how the ML-Series card is configured. The **show interface** command shows the maximum bandwidth of a port (for example, BW 100000 Kbit). The sum of all bandwidth and priority actions applied to the interface, plus the cos priority-mcast bandwidth, is not allowed to exceed the maximum bandwidth of the port.

Table 11-2 Traffic Policy Commands (continued)

Command	Purpose
`Router (config-pmap-c)# police cir-rate-bps normal-burst-byte [max-burst-byte] [pir pir-rate-bps] [conform-action {set-cos-transmit | transmit | drop}] [exceed-action {set-cos-transmit | drop}] [violate-action {set-cos-transmit | drop}]`	Defines a policer for the currently selected class when the policy map is applied to input. Policing is supported only on ingress, not on egress. • For *cir-rate-bps,* specify the average committed information rate (cir) in bits per second (bps). The range is 96000 to 800000000. • For *normal-burst-byte,* specify the cir burst size in bytes. The range is 8000 to 64000. • (Optional) For *maximum-burst-byte*, specify the peak information rate (pir) burst in bytes. The range is 8000 to 64000. • (Optional) For *pir-rate-bps*, specify the average pir traffic rate in bps where the range is 96000 to 800000000. • (Optional) Conform action options are: – Set a CoS priority value and transmit – Transmit packet (default) – Drop packet • (Optional) Exceed action options are: – Set a CoS value and transmit – Drop packet (default) • (Optional) The violate action is only valid if pir is configured. Violate action options are: – Set a CoS value and transmit – Drop packet (default)

Table 11-2 Traffic Policy Commands (continued)

Command	Purpose
`ML_Series (config-pmap-c)# priority kbps`	Specifies low latency queuing for the currently selected class. This command can only be applied to an output. When the policy-map is applied to an output, an output queue with strict priority is created for this class. The only valid rate choice is in kilobits per second. **Note** This **priority** command does not apply to the default class. **Note** When using the priority action, the traffic in that class is given a 100 percent CIR, regardless of the rate entered as the priority rate. To ensure that other bandwidth commitments are met for the interface, a policer must be configured on the input of all interfaces that might deliver traffic to this output class, limiting the peak rate to the priority rate entered. **Note** The true configurable bandwidth in kilobits per second is per port and depends on how the ML-Series card is configured. The **show interface** command shows the maximum bandwidth of a port (for example, BW 100000 Kbit). The sum of all bandwidth and priority actions applied to the interface, plus the cos priority-mcast bandwidth, is not allowed to exceed the maximum bandwidth of the port.
`ML_Series (config-pmap-c)# set cos cos-value`	Specifies a CoS value or values to associate with the packet. The number is in the range from 0 to 7. This command can only be used in a policy-map applied to an output. It specifies the VLAN CoS priority to set for the outbound packets in the currently selected class. If QinQ is used, the top-level VLAN tag is marked. If outbound packets have no VLAN tag, the action has no effect. This action is applied to the packet after any set-cos action done by a policer, and therefore overrides the CoS set by a policer action. If a packet is marked by the policer and forwarded out through an interface that also has a set-cos action assigned for the traffic class, the value specified by the police action takes precedence in setting the IEEE 802.1p CoS field. This command also sets the CoS value in the RPR header for packets exiting the ML-Series on the RPR interface.

Attaching a Traffic Policy to an Interface

Use the **service-policy** interface configuration command to attach a traffic policy to an interface and to specify the direction in which the policy should be applied (either on packets coming into the interface or packets leaving the interface). Only one traffic policy can be applied to an interface in a given direction.

Use the **no** form of the command to detach a traffic policy from an interface. The **service-policy** command syntax is as follows:

service-policy {**input** | **output**} *policy-map-name*
no service-policy {**input** | **output**} *policy-map-name*

To attach a traffic policy to an interface, perform the following procedure in global configuration mode:

	Command	Purpose
Step 1	`ML_Series(config)# interface interface-id`	Enters interface configuration mode, and specifies the interface to apply the policy map. Valid interfaces are limited to physical Ethernet and packet-over-SONET (POS) interfaces. **Note** Policy maps cannot be applied to SPR interfaces, subinterfaces, port channel interfaces, or Bridge Group Virtual Interfaces (BVIs).
Step 2	`ML_Series(config-if)# service-policy output policy-map-name`	Specifies the name of the traffic policy to be attached to the output direction of an interface. The traffic policy evaluates all traffic leaving that interface.
Step 3	`ML_Series(config-if)# service-policy input policy-map-name`	Specifies the name of the traffic policy to be attached to the input direction of an interface. The traffic policy evaluates all traffic entering that interface.

Configuring CoS-Based QoS

The global **cos commit** *cos-value* command allows the ML-Series card to base the QoS treatment for a packet coming in on a network interface on the attached CoS value, rather than on a per-customer-queue policer.

CoS-based QoS is applied with a single global **cos commit** *cos-value* command, as shown in Table 11-3:

Table 11-3 CoS Commit Command

Command	Purpose
`ML_Series(config)# cos-commit cos-value`	Labels packets that come in with a CoS equal to or higher than the *cos-value* as CIR and packets with a lower CoS as DE.

Monitoring and Verifying QoS Configuration

After configuring QoS on the ML-Series card, the configuration of class maps and policy maps can be viewed through a variety of **show** commands. To display the information relating to a traffic class or traffic policy, use one of the following commands in EXEC mode, as needed. Table 11-4 describes the commands that are related to QoS status.

Table 11-4 Commands for QoS Status

Command	Purpose
`ML_Series# show class-map name`	Displays the traffic class information of the user-specified traffic class.
`ML_Series# show policy-map`	Displays all configured traffic policies.

Table 11-4 Commands for QoS Status (continued)

Command	Purpose
ML_Series# **show policy-map** *name*	Displays the user-specified policy map.
ML_Series# **show policy-map interface** *interface*	Displays configurations of all input and output policies attached to an interface. Statistics displayed with this command are unsupported and show zero.

Example 11-1 shows examples of the QoS commands.

Example 11-1 QoS Status Command Examples

```
ML_Series# show class-map
 Class Map match-any class-default (id 0)
   Match any
 Class Map match-all policer (id 2)
   Match ip precedence 0

ML_Series# show policy-map
 Policy Map police_f0
  class  policer
   police 1000000 10000 conform-action transmit exceed-action drop

ML_Series# show policy-map interface

 FastEthernet0

  service-policy input: police_f0

    class-map: policer (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      match: ip precedence 0

    class-map: class-default (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      match: any
        0 packets, 0 bytes
        5 minute rate 0 bps
```

QoS Configuration Examples

This section provides the specific command and network configuration examples:

- Traffic Classes Defined Example
- Traffic Policy Created Example
- class-map match-any and class-map match-all Commands Example
- match spr1 Interface Example
- ML-Series VoIP Example
- ML-Series Policing Example
- ML-Series CoS-Based QoS Example

Traffic Classes Defined Example

Example 11-2 shows how to create a class map called class1 that matches incoming traffic entering interface fastethernet0.

Example 11-2 Class Interface Command Example

```
ML_Series(config)# class-map class1
ML_Series(config-cmap)# match input-interface fastethernet0
```

Example 11-3 shows how to create a class map called class2 that matches incoming traffic with IP-precedence values of 5, 6, and 7.

Example 11-3 Class IP-Precedence Command Example

```
ML_Series(config)# class-map match-any class2
ML_Series(config-cmap)# match ip precedence 5 6 7
```

Note If a class-map contains a match rule that specifies multiple values, such as 5 6 7 in this example, then the class-map must be match-any, not the default match-all. Without the match-any class-map, an error message is printed and the class is ignored. The supported commands that allow multiple values are **match cos**, **match ip precedence**, and **match ip dscp**.

Example 11-4 shows how to create a class map called class3 that matches incoming traffic based on bridge group 1.

Example 11-4 Class Map Bridge Group Command Example

```
ML_Series(config)# class-map class3
ML_Series(config-cmap)# match bridge-group 1
```

Traffic Policy Created Example

In Example 11-5, a traffic policy called policy1 is defined to contain policy specifications, including a bandwidth allocation request for the default class and two additional classes—class1 and class2. The match criteria for these classes were defined in the traffic classes (see the "Creating a Traffic Class" section on page 11-10).

Example 11-5 Traffic Policy Created Example

```
ML_Series(config)# policy-map policy1
ML_Series(config-pmap)# class class-default
ML_Series(config-pmap-c)# bandwidth 1000
ML_Series(config-pmap)# exit

ML_Series(config-pmap)# class class1
ML_Series(config-pmap-c)# bandwidth 3000
ML_Series(config-pmap)# exit

ML_Series(config-pmap)# class class2
ML_Series(config-pmap-c)# bandwidth 2000
ML_Series(config-pmap)# exit
```

class-map match-any and class-map match-all Commands Example

This section illustrates the difference between the **class-map match-any** command and the **class-map match-all** command. The **match-any** and **match-all** options determine how packets are evaluated when multiple match criteria exist. Packets must either meet all of the match criteria (**match-all**) or one of the match criteria (**match-any**) in order to be considered a member of the traffic class.

Example 11-6 shows a traffic class configured with the **class-map match-all** command.

Example 11-6 Class Map Match All Command Example

```
ML_Series(config)# class-map match-all cisco1
ML_Series(config-cmap)# match cos 1
ML_Series(config-cmap)# match bridge-group 10
```

If a packet arrives with a traffic class called cisco1 configured on the interface, the packet is evaluated to determine if it matches the cos 1 and bridge group 10. If both of these match criteria are met, the packet matches traffic class cisco1.

In a traffic class called cisco2, the match criteria are evaluated consecutively until a successful match criterion is located. The packet is first evaluated to the determine whether cos 1 can be used as a match criterion. If cos 1 can be used as a match criterion, the packet is matched to traffic class cisco2. If cos 1 is not a successful match criterion, then bridge-group 10 is evaluated as a match criterion. Each matching criterion is evaluated to see if the packet matches that criterion. When a successful match occurs, the packet is classified as a member of traffic class cisco2. If the packet matches none of the specified criteria, the packet is classified as a member of the traffic class.

Note that the **class-map match-all** command requires that all of the match criteria must be met in order for the packet to be considered a member of the specified traffic class (a logical AND operator). In the example, cos 1 AND bridge group 10 have to be successful match criteria. However, only one match criterion must be met for the packet in the **class-map match-any** command to be classified as a member of the traffic class (a logical OR operator).

Example 11-7 shows a traffic class configured with the **class-map match-any** command. In the example, cos 1 OR bridge group 10 OR ip dscp 5 has to be successful match criteria.

Example 11-7 Class Map Match Any Command Example

```
ML_Series(config)# class-map match-any cisco2
ML_Series(config-cmap)# match cos 1
ML_Series(config-cmap)# match bridge-group 10
ML_Series(config-cmap)# match ip dscp 5
```

match spr1 Interface Example

In Example 11-8, the SPR interface is specified as a parameter to the **match input-interface** CLI when defining a class-map.

Example 11-8 Class Map SPR Interface Command Example

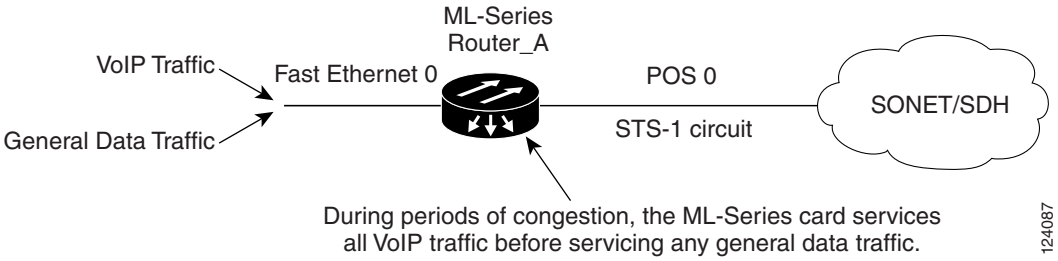
```
ML_Series(config)# class-map spr1-cos1
ML_Series(config-cmap)# match input-interface spr1
ML_Series(config-cmap)# match cos 1
ML_Series(config-cmap)# end
ML_Series# sh class-map spr1-cos1
 Class Map match-all spr1-cos1 (id 3)
```

```
Match input-interface SPR1
Match cos  1
```

ML-Series VoIP Example

Figure 11-7 shows an example of ML-Series voice-over- IP (VoIP). The associated commands are provided in Example 11-9.

Figure 11-7 ML-Series VoIP Example



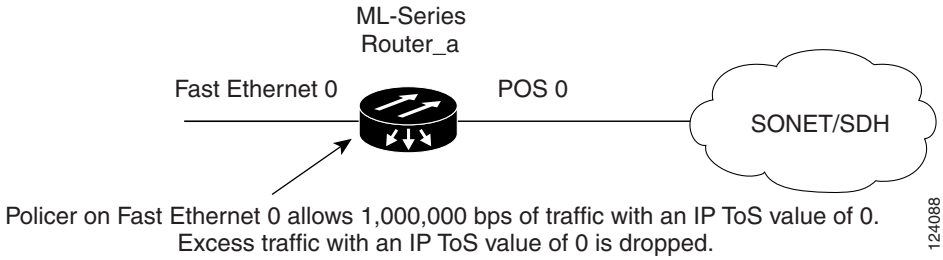
Example 11-9 ML-Series VoIP Commands

```
Router(config)# class-map match-all voip
Router(config-cmap)# match ip precedence 5
Router(config-cmap)# exit
Router(config)# class-map match-any default
Router(config-cmap)# match any
Router(config-cmap)# exit
Router(config)# policy-map pos0
Router(config-pmap)# class default
Router(config-pmap-c)# bandwidth   1000
Router(config-pmap-c)# class voip
Router(config-pmap-c)# priority 1000
Router(config-pmap-c)# interface FastEthernet0
Router(config-if)# ip address 1.1.1.1 255.255.255.0
Router(config-if)# interface POS0
Router(config-if)# ip address 2.1.1.1 255.255.255.0
Router(config-if)# service-policy output pos0
Router(config-if)# crc 32
Router(config-if)# no cdp enable
```

ML-Series Policing Example

Figure 11-8 shows an example of ML-Series policing. The example shows how to configure a policer that restricts traffic with an IP precedence of 0 to 1,000,000 bps. The associated code is provided in Example 11-10.

Figure 11-8 ML-Series Policing Example



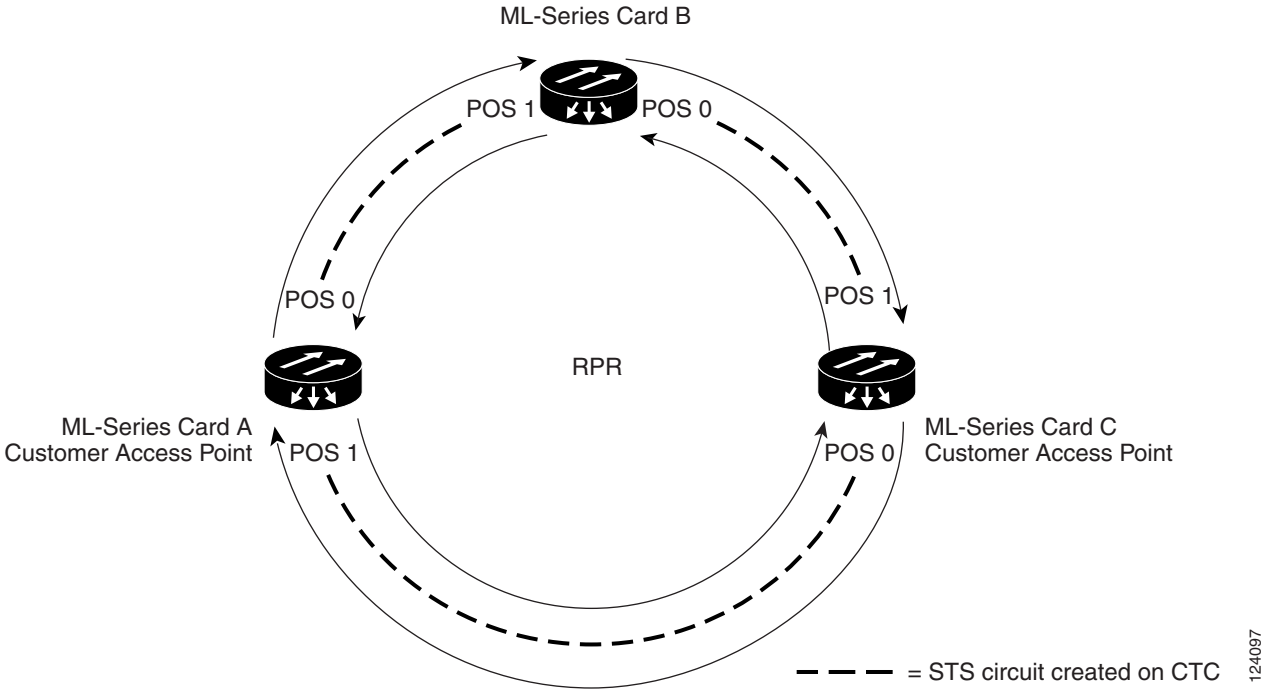
Example 11-10 ML-Series Policing Commands

```
Router(config)# class-map match-all policer
Router(config-cmap)# match ip precedence 0
Router(config-cmap)# exit
Router(config)# policy-map police_f0
Router(config-pmap)# class policer
Router(config-pmap-c)# police 1000000 10000 conform-action transmit exceed-action drop
Router(config-pmap-c)# interface FastEthernet0
Router(config-if)# service-policy input police_f0
```

ML-Series CoS-Based QoS Example

Figure 11-9 shows an example of ML-Series CoS-based QoS. The associated code is provided in the examples that follow the figure. The CoS example assumes that the ML-Series cards are configured into an RPR and that the ML-Series card POS ports are linked by point-to-point SONET circuits. ML-Series Card A and ML-Series Card C are customer access points. ML-Series Card B is not a customer access point. For more information on configuring RPR, see Chapter 14, "Configuring Resilient Packet Ring on the ML-Series Card."

Figure 11-9 ML-Series CoS Example



Example 11-11 shows the code used to configure ML-Series Card A in Figure 11-9.

Example 11-11 ML-Series Card A Configuration (Customer Access Point)

```
ML_Series_A(config)# cos commit 2
ML_Series_A(config)# policy-map Fast5_in
ML_Series_A(config-pmap)# class class-default
ML_Series_A(config-pmap-c)# police 5000 8000 8000 pir 10000 conform-action
set-cos-transmit 2 exceed-action set-cos-transmit 1 violate-action drop
```

Example 11-12 shows the code used to configure ML-Series Card B in Figure 11-9.

Example 11-12 ML-Series Card B Configuration (Not a Customer Access Point)

```
ML_Series_B(config)# cos commit 2
```

Example 11-13 shows the code used to configure ML-Series Card C in Figure 11-9.

Example 11-13 ML-Series Card C Configuration (Customer Access Point)

```
ML_Series_B(config)# cos commit 2
ML_Series_B(config)# policy-map Fast5_in
ML_Series_B(config-pmap)# class class-default
ML_Series_B(config-pmap-c)# police 5000 8000 8000 pir 10000 conform-action
set-cos-transmit 2 exceed-action set-cos-transmit 1 violate-action drop
```

Understanding Multicast QoS and Multicast Priority Queuing

ML-Series card QoS supports the creation of two priority classes for multicast traffic in addition to the default multiclass traffic class. Creating a multicast priority queuing class of traffic configures the ML-Series card to recognize an existing CoS value in ingressing multicast traffic for priority treatment.

The multicast priority queuing CoS match is based on the “internal" CoS value of each packet. This value is normally the same as the egress CoS value (after policer marking if enabled) but differs in two cases. The “internal” CoS value is not the same as the egress value when dot1q-tunneling is used. With dot1q-tunneling, the internal CoS value is always the value of the outer tag CoS, both when entering the dot1q tunnel and leaving the dot1q tunnel. The “internal” CoS value is also not the same as the egress value if a packet is transported over a VLAN, and the VLAN tag is removed on egress to send the packet untagged. In this case, the internal CoS is the CoS of the removed tag (including ingress policing and marking if enabled).

The **cos priority-mcast** command does not modify the CoS of the multicast packets but only the bandwidth allocation for the multicast priority queuing class. The command guarantees a minimum amount of bandwidth and is queued separately from the default multicast/broadcast queue.

Creating a multicast priority queuing class allows for special handling of certain types of multiclass traffic. This is especially valuable for multicast video distribution and service provider multicast traffic. For example, a service provider might want to guarantee the protection of their own multicast management traffic. To do this, they could create a multicast priority queuing class on the ML-Series card for the CoS value of the multicast management traffic and guarantee its minimum bandwidth. For multicast video distribution, a multicast priority queuing class on the ML-Series card for the CoS value of the multicast video traffic enables networks to efficiently manage multicast video bandwidth demands on a network shared with VoIP and other Ethernet services.

Note Multicast priority queuing traffic uses port-based load-balancing over RPR and EtherChannel. Default multicast traffic is load-balanced over RPR, but not over EtherChannel.

Note Multicast priority queuing bandwidth should not be oversubscribed for sustained periods with traffic from multiple sources. This can result in reduced multicast priority queuing throughput.

Default Multicast QoS

Default multicast traffic is any multicast traffic (including flooded traffic) that is not classified as multicast priority queuing. The default multicast class also includes broadcast data traffic, control traffic, L2 protocol tunneling, and flooding traffic of the unknown MAC during MAC learning.

With no QoS configured (no multicast priority queuing and no output policy map) on the ML-Series card, the default multicast bandwidth is a 10 percent minimum of the total bandwidth.

When bandwidth is allocated to multicast priority queuing but no output policy map is applied, the default multicast congestion bandwidth is a minimum of 10 percent of the bandwidth that is not allocated to multicast priority queuing.

When an output policy-map is applied to an interface, default multicast and default unicast share the minimum bandwidth assigned to the default class. This default class is also known as the match-any class. The minimum bandwidth of default multicast is 10 percent of the total default class bandwidth.

Multicast Priority Queuing QoS Restrictions

The following restrictions apply to multicast priority queuing QoS:

- The bandwidth allocation and utilization configured for multicast priority queuing traffic is global and applies to all the ports on the ML-Series card, both POS and Fast Ethernet, regardless of whether these ports carry multicast priority queuing traffic. The rate of traffic can be reduced for all ports on the ML-Series card when this feature is configured. Default multicast traffic uses bandwidth only on the ports where it egresses, not globally like multicast priority queuing.
- Multicast priority queuing QoS is supported only for Layer 2 bridging.
- The ML-Series card supports a maximum of two multicast priority queuing classes.
- Unlike the rest of the ML-Series card QoS, multicast priority queuing QoS is not part of the Cisco IOS MQC.
- Priority-mcast bandwidth allocation is per port.

Configuring Multicast Priority Queuing QoS

To configure a priority class for multicast traffic, use the global configuration **cos priority-mcast** command defined in Table 11-5.

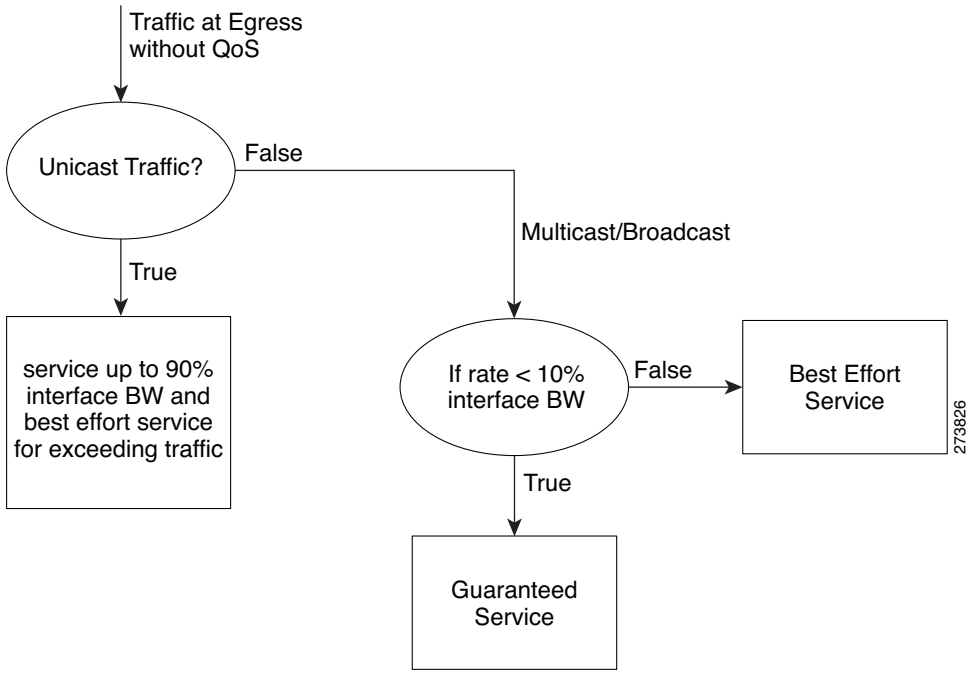
Table 11-5 CoS Multicast Priority Queuing Command

Command	Purpose
`Router (config)# [no] cos priority-mcast cos-value {bandwidth-kbps | mbps bandwidth-mbps | percent percent}`	*Creates a priority class of multicast traffic based on a multicast CoS value and* specifies a minimum bandwidth guarantee to a traffic class in periods of congestion. *cos-value* specifies the CoS value of multicast packets which will be given the bandwidth allocation. Matches only a single CoS of traffic (not a range). Supported CoS range is 0 to 7. A minimum bandwidth guarantee can be specified in kbps, in mbps, or by a percentage of the overall available bandwidth. Valid choices for the ML-Series card are: • Rate in kilobits per second • Rate in megabits per second • Percent of total available port bandwidth (1 to 100) Reentering the command with the same *cos-value* but a different bandwidth rate will modify the bandwidth of the existing class. Reentering the command with a different *cos-value* creates a separate multicast priority queuing class with a maximum of two multicast priority queuing classes. The **no** form of this command removes the multicast priority queuing class. **Note** The true configurable bandwidth in kilobits or megabits per second is per port and depends on how the ML-Series card is configured. The **show interface** command shows the maximum bandwidth of a port (for example, BW 100000 Kbit). The sum of all bandwidth and priority actions applied to the interface, plus the cos priority-mcast bandwidth, is not allowed to exceed the maximum bandwidth of the port. **Note** Attempting to configure a priority-mcast bandwidth that exceeds the true configurable bandwidth on any port will cause the priority-mcast configuration change to fail, and the multicast priority queuing bandwidth guarantee will not be changed.

QoS not Configured on Egress

The QoS bandwidth allocation of multicast and broadcast traffic is handled separately from unicast traffic. On each interface, the aggregate multicast and broadcast traffic are given a fixed bandwidth commit of 10% of the interface bandwidth. This is the optimum bandwidth that can be provided for traffic exceeding 10% of the interface bandwidth.

Figure 11-10 QoS not Configured on Egress



ML-Series Egress Bandwidth Example

This section explains with examples the utilization of bandwidth across different queues with or without Priority Multicast.

Case 1: QoS with Priority and Bandwidth Configured Without Priority Multicast

Strict Priority Queue is always serviced first. The remaining interface bandwidth is utilized to service other configured traffic.

In the following example, after servicing unicast `customer_voice` traffic, the remaining interface bandwidth is utilized for other WRR queues such as `customer_core_traffic`, `customer_data`, and `class-default` in the ratio of 1:3:5.

At any given time, the sum of the bandwidth assigned cannot exceed the interface bandwidth (in kbps). The bandwidth share allocated to `class-default` will be utilized by default unicast traffic (in this example, unicast traffic with CoS values other than 2, 5, 7) and all multicast/broadcast traffic (all CoS values). The default unicast and all multicast/broadcast traffic will be serviced in the ratio of 9:1.

For example, if 18x bandwidth is available after servicing priority unicast traffic (CoS 5), then the remaining bandwidth will be allocated as follows:

Unicast traffic with CoS 2 : 2x

Unicast traffic with CoS 7: 6x

Unicast default (without CoS 2, CoS 5, CoS 7): 9x

All multicast/broadcast (any CoS value): 1x

Example 11-14 QoS with Priority and Bandwidth Configured without Priority Multicast

```
!
class-map match-all customer_voice
  match cos  5
class-map match-all customer_data
  match cos  7
class-map match-all customer_core_traffic
  match cos  2
!
!
policy-map policy_egress_bandwidth
  class customer_core_traffic
    bandwidth   1000
  class customer_voice
    priority 1000
  class customer_data
    bandwidth   3000
  class class-default
    bandwidth   5000
!
!
interface POS0
 no ip address
  crc 32
 service-policy output policy_egress_bandwidth
!
```

Case 2: QoS with Priority and Bandwidth Configured with Priority Multicast

In this case, only multicast traffic of CoS 3 is allocated a guaranteed bandwidth. This multicast traffic will now participate in the queue along with other WRR queues. After servicing the `customer_voice` traffic, the remaining interface bandwidth is utilized for WRR queues, such as `customer_core_traffic`, `customer_data`, `class-default`, and multicast CoS 3 traffic in the ratio of 1:3:5:2.

At any given time, the sum of the bandwidth assigned cannot exceed the interface bandwidth (in kbps).

Example 11-15 QoS with Priority and Bandwidth configured with Priority Multicast

```
cos priority-mcast 3 2000
!
class-map match-all customer_voice
  match cos  5
class-map match-all customer_data
  match cos  7
class-map match-all customer_core_traffic
  match cos  2
!
!
```

```
policy-map policy_egress_bandwidth
  class customer_core_traffic
    bandwidth   1000
  class customer_voice
    priority 1000
  class customer_data
    bandwidth   3000
  class class-default
    bandwidth   5000
!
!
interface POS0
 no ip address
  crc 32
 service-policy output policy_egress_bandwidth
!
```

Understanding CoS-Based Packet Statistics

Note For IEEE 802.1Q (QinQ) enabled interfaces, CoS accounting is based only on the CoS value of the outer metro tag imposed by the service provider. The CoS value inside the packet sent by the customer network is not considered for CoS accounting.

Enhanced performance monitoring displays per-CoS packet statistics on the ML-Series card interfaces when CoS accounting is enabled. CoS-based traffic utilization is displayed at the Fast Ethernet interface or subinterface (VLAN) level, or at the POS interface level. It is not displayed at the POS subinterface level. RPR statistics are not available at the SPR interface level, but statistics are available for the individual POS ports that make up the SPR interface. EtherChannel (port-channel) and BVI statistics are available only at the member port level. Table 11-6 shows the types of statistics available at specific interfaces.

Table 11-6 Packet Statistics on ML-Series Card Interfaces

Statistics Collected	Fast Ethernet Interface	Fast Ethernet Subinterface (VLAN)	POS Interface	POS Subinterface
Input—Packets and Bytes	Yes	Yes	No	No
Output—Packets and Bytes	Yes	Yes	No	No
Drop Count—Packets and Bytes[1]	Yes	No	Yes	No

1. Drop counts only include discards caused by output congestion and are counted at the output interface.

CoS-based packet statistics are available through the Cisco IOS command-line interface (CLI) and Simple Network Management Protocol (SNMP), using an extension of the CISCO-PORT-QOS MIB. They are not available through Cisco Transport Controller (CTC).

Configuring CoS-Based Packet Statistics

Note For IEEE 802.1Q (QinQ) enabled interfaces, CoS accounting is based only on the CoS value of the outer metro tag imposed by the service provider. The CoS value inside the packet sent by the customer network is not considered for CoS accounting.

To enable CoS-based packet statistics on an interface, use the interface configuration level command defined in Table 11-7.

Table 11-7 CoS-Based Packet Statistics Command

Command	Purpose
`ML_Series(config-if)# cos accounting`	Enables CoS-based packet statistics to be recorded at the specific interface and for all the subinterfaces of that interface. This command is supported only in interface configuration mode and not subinterface configuration mode. The **no** form of the command disables the statistics.

After configuring CoS-based packet statistics on the ML-Series card, the statistics can be viewed through a variety of **show** commands. To display this information, use one of the commands in Table 11-8 in EXEC mode.

Table 11-8 Commands for CoS-Based Packet Statistics

Command	Purpose
`ML_Series# show interface type number cos`	Displays the CoS-based packet statistics available for an interface.
`ML_Series# show interface type number.subinterface-number cos`	Displays the CoS-based packet statistics available for a FastEthernet subinterface. POS subinterfaces are not eligible.

Example 11-16 shows examples of these commands.

Example 11-16 Commands for CoS-Based Packet Statistics Examples

```
ML_Series# show interface fastethernet 0.5 cos
FastEthernet0.5
    Stats by Internal-Cos
    Input: Packets         Bytes
        Cos 0: 31          2000
        Cos 1:
        Cos 2: 5           400
        Cos 3:
        Cos 4:
        Cos 5:
        Cos 6:
        Cos 7:
    Output: Packets        Bytes
        Cos 0: 1234567890  1234567890
        Cos 1: 31          2000
        Cos 2:
```

```
        Cos 3:
        Cos 4:
        Cos 5:
        Cos 6: 10          640
        Cos 7:

ML_Series# show interface fastethernet 0 cos
FastEthernet0
    Stats by Internal-Cos
    Input: Packets         Bytes
        Cos 0: 123         3564
        Cos 1:
        Cos 2: 3           211
        Cos 3:
        Cos 4:
        Cos 5:
        Cos 6:
        Cos 7:
    Output: Packets        Bytes
        Cos 0: 1234567890  1234567890
        Cos 1: 3           200
        Cos 2:
        Cos 3:
        Cos 4:
        Cos 5:
        Cos 6: 1           64
        Cos 7:
    Output: Drop-pkts      Drop-bytes
        Cos 0: 1234567890  1234567890
        Cos 1:
        Cos 2:
        Cos 3:
        Cos 4:
        Cos 5: 1               64
        Cos 6: 10              640
        Cos 7:

ML_Series# show interface pos0 cos
    POS0
    Stats by Internal-Cos
    Output: Drop-pkts          Drop-bytes
        Cos 0: 12              1234
        Cos 1: 31              2000
        Cos 2:
        Cos 3:
        Cos 4:
        Cos 5:
        Cos 6: 10              640
        Cos 7:
```

Understanding IP SLA

Cisco IP SLA, formerly known as the Cisco Service Assurance Agent, is a Cisco IOS feature to assure IP service levels. Using IP SLA, service provider customers can measure and provide service level agreements, and enterprise customers can verify service levels, verify outsourced service level agreements, and understand network performance for new or existing IP services and applications. IP SLAs use unique service level assurance metrics and methodology to provide highly accurate, precise service level assurance measurements.

Depending on the specific IP SLAs operation, statistics of delay, packet loss, jitter, packet sequence, connectivity, path, server response time, and download time are monitored within the Cisco device and stored in both CLI and SNMP MIBs. The packets have configurable IP and application layer options such as source and destination IP address, User Datagram Protocol (UDP)/TCP port numbers, a type of service (ToS) byte (including Differentiated Services Code Point [DSCP] and IP Prefix bits), Virtual Private Network (VPN) routing/forwarding instance (VRF), and URL web address.

IP SLAs uses generated traffic to measure network performance between two networking devices such as routers. IP SLAs starts when the IP SLAs device sends a generated packet to the destination device. After the destination device receives the packet, and depending on the type of IP SLAs operation, the device will respond with time-stamp information for the source to make the calculation on performance metrics. An IP SLAs operation is a network measurement to a destination in the network from the source device using a specific protocol such as UDP for the operation.

Because IP SLA is accessible using SNMP, it also can be used in performance monitoring applications for network management systems (NMSs) such as CiscoWorks2000 (CiscoWorks Blue) and the Internetwork Performance Monitor (IPM). IP SLA notifications also can be enabled through Systems Network Architecture (SNA) network management vector transport (NMVT) for applications such as NetView.

For general IP SLA information, refer to the Cisco IOS IP Service Level Agreements technology page at http://www.cisco.com/warp/public/732/Tech/nmp/ipsla. For information on configuring the Cisco IP SLA feature, see the "Network Monitoring Using Cisco Service Assurance Agent" chapter of the *Cisco IOS Configuration Fundamentals Configuration Guide, Release 12.2*. at: http://www.cisco.com/en/US/products/sw/iosswrel/ps1835/products_configuration_guide_chapter09186a008030c773.html.

IP SLA on the ML-Series

The ML-Series card has a complete IP SLA Cisco IOS subsystem and offers all the normal features and functions available in Cisco IOS Release 12.2S. It uses the standard IP SLA Cisco IOS CLI commands. The SNMP support will be equivalent to the support provided in the IP SLA subsystem 12.2(S), which is the rttMon MIB.

IP SLA Restrictions on the ML-Series

The ML-Series card supports only features in the Cisco IOS 12.2S branch. It does not support functions available in future Cisco IOS versions, such as the IP SLA accuracy feature or the enhanced Cisco IOS CLI support with updated IP SLA nomenclature.

Other restrictions are:

- Setting the CoS bits is supported, but set CoS bits are not honored when leaving or entering the CPU when the sender or responder is an ONS 15454, ONS 15454 SDH, ONS 15310-CL, or ONS 15310-MA platform. Set CoS bits are honored in intermediate ONS nodes.
- On RPR, the direction of the data flow for the IP SLA packet might differ from the direction of customer traffic.
- The system clock on the ML-Series card synchronizes with the clock on the TCC2/TCC2P card. Any NTP server synchronization is done with the TCC2/TCC2P card clock and not with the ML-Series card clock.

- The average Round Trip Time (RTT) measured on an ML-Series IP SLA feature is more than the actual data path latency. In the ML-Series cards, IP SLA is implemented in the software. The IP SLA messages are processed in the CPU of the ML-Series card. The latency time measured includes the network latency and CPU processing time. For very accurate IP SLA measurements, it is recommended that a Cisco Router or Switch be used as an external probe or responder to measure the RTT of the ML-Series cards in a network.

CHAPTER 12

Configuring the Switching Database Manager on the ML-Series Card

This chapter describes the switching database manager (SDM) features built into the ML-Series card and contains the following major sections:

Understanding the SDM

The ONS 15310-CL and ONS 15310-MA ML-Series card features high-speed forwarding. The ML-Series card does Layer 2 MAC address lookups through a hash table. Quality of service (QoS) classifier lookup are done in software and all other lookups are supported by the main policy engine. The ONS 15310-CL and ONS 15310-MA ML-Series card does not use external ternary content-addressable memory (TCAM) like the ONS 15454 ML-Series card.

The SDM is the software subsystem that manages the switching information. It organizes the switching information into application-specific regions and configures the size of these application regions. SDM enables exact-match and longest-match address searches, which result in high-speed forwarding.

A location index is associated with each packet forwarded and conveyed to the forwarding engine. The forwarding engine uses this location index to derive information associated with each forwarded packet.

Understanding SDM Regions

SDM partitions multiple application-specific regions and interacts with the individual application control layers to store switching information. The regions share the total available space. SDM consists of the following types of regions:

- Exact-match region—The exact-match region consists of entries for multiple application regions such as IP adjacencies.

- Longest-match region—Each longest-match region consists of multiple buckets or groups of Layer 3 address entries organized in decreasing order by mask length. All entries within a bucket share the same mask value and key size. The buckets can change their size dynamically by borrowing address entries from neighboring buckets. Although the size of the whole application region is fixed, you can reconfigure it.
- Weighted-exact-match region—The weighted-exact-match region consists of exact-match-entries with an assigned weight or priority. For example, with QoS, multiple exact match entries might exist, but some have priority over others. The weight is used to select one entry when multiple entries match.

Table 12-1 lists default partitioning for each application region.

Table 12-1 Default Partitioning by Application Region

Application Region	Lookup Type	Key Size	Default Size
IP Adjacency	Exact-match	64 bits	300 (shared)
IP Prefix	Longest-match	64 bits	300 (shared)
QoS Classifiers	Weighted exact-match	64 bits	300 (shared)
IP VRF Prefix	Longest prefix match	64 bits	300 (shared)
IP Multicast	Longest prefix match	64 bits	300 (shared)
MAC Addr	Longest prefix match	64 bits	8192
Access List	Weighted exact match	64 bits	300 (shared)

Configuring SDM

This section describes SDM region size and access control list (ACL) size configuration. The commands described in this section are unique to the switching software. Configuration changes take place immediately on the ML-100T-8 card.

Configuring SDM Regions

To configure SDM maximum size for each application region, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	`ML_Series(config)# sdm size` *region-name number-of-entries*	Configures the maximum number of entries for an SDM region.
Step 2	`ML_Series(config)# end`	Exits to privileged EXEC mode.

An example of this is shown in Example 12-1.

Example 12-1 Limiting the IP-Prefix Region to 2K Entries

```
ML_Series # configure terminal
ML_Series(config)# sdm size ip-prefix 200
ML_Series(config)# end
```

Configuring Access Control List Size in TCAM

The default maximum size of the ACL is 300 64-bit entries. You can enter the **sdm access-list** command to change the maximum ACL database size, as shown in Table 12-2.

Table 12-2 Partitioning the TCAM Size for ACLs

Task	Command
sdm access-list *number-entries*	Sets the name of the application region for which you want to configure the size. You can enter the size as an absolute number of entries.

An example of this is shown in Example 12-2.

Example 12-2 Configuring Entries for the ACL Region in TCAM

```
ML_Series# configure terminal
ML_Series(config)# sdm access-list 100
ML_Series(config)# end
```

Monitoring and Verifying SDM

To display the number of available TCAM entries, enter the **show sdm size** command from global configuration mode:

```
ML_Series # show sdm size
Active Switching Database Region Maximum Sizes :
    IP Adjacency        : 300     64-bit entries
    IP Prefix           : 300     64-bit entries
    QoS Classifiers     : 300     64-bit entries
    IP VRF Prefix       : 300     64-bit entries
    IP Multicast        : 300     64-bit entries
    MAC Addr            : 8192    64-bit entries
    Access List         : 300     64-bit entries
```

CHAPTER 13

Configuring Access Control Lists on the ML-Series Card

This chapter describes the access control list (ACL) features built into the ML-Series card and contains the following major sections:

Understanding ACLs

ACLs provide network control and security, allowing you to filter packet flow into or out of ML-Series interfaces. ACLs, which are sometimes called filters, allow you to restrict network use by certain users or devices. ACLs are created for each protocol and are applied on the interface for either inbound or outbound traffic. ACLs do not apply to outbound control plane traffic. Only one ACL filter can be applied per direction per subinterface.

When creating ACLs, you define criteria to apply to each packet processed by the ML-Series card; the ML-Series card decides whether to forward or block the packet based on whether or not the packet matches the criteria in your list. Packets that do not match any criteria in your list are automatically blocked by the implicit “deny all traffic” criteria statement at the end of every ACL.

ML-Series ACL Support

Both control-plane and data-plane ACLs are supported on the ML-Series card:

- Control-plane ACLs: ACLs used to filter control data that is processed by the CPU of the ML-Series card (for example, distribution of routing information, Internet Group Membership Protocol (IGMP) joins, and so on).
- Data-plane ACLs: ACLs used to filter user data being routed or bridged through the ML Series in hardware (for example, denying access to a host, and so on). These ACLs are applied to an interface in the input or output direction using the **ip access-group** command.

The following apply when using data-plane ACLs on the ML-Series card:

- ACLs are supported on all interface types, including bridged interfaces.
- Reflexive and dynamic ACLs are not supported on the ML-Series card.

- Access violations accounting is not supported on the ML-Series card.
- ACL logging is supported only for packets going to the CPU, not for switched packets.
- IP standard ACLs applied to bridged egress interfaces are not supported in the data-plane. When bridging, ACLs are only supported on ingress.

IP ACLs

The following ACL styles for IP are supported:

- Standard IP ACLs: These use source addresses for matching operations.
- Extended IP ACLs: (Control plane only) These use source and destination addresses for matching operations and optional protocol type and port numbers for finer granularity of control.
- Named ACLs: These use source addresses for matching operations.

Note By default, the end of the ACL contains an implicit deny statement for everything if it did not find a match before reaching the end. With standard ACLs, if you omit the mask from an associated IP host address ACL specification, 0.0.0.0 is assumed to be the mask.

After creating an ACL, you must apply it to an interface, as shown in the "Applying the ACL to an Interface" section on page 13-4.

Named IP ACLs

You can identify IP ACLs with a name, but it must be an alphanumeric string. Named IP ACLs allow you to configure more IP ACLs in a router than if you used numbered ACLs. If you identify your ACL with an alphabetic rather than a numeric string, the mode and command syntax are slightly different.

Consider the following before configuring named ACLs:

- A standard ACL and an extended ACL cannot have the same name.
- Numbered ACLs are also available, as described in the "Creating Numbered Standard and Extended IP ACLs" section on page 13-3.

User Guidelines

Keep the following in mind when you configure IP network access control:

- You can program ACL entries into Ternary Content Addressable Memory (TCAM).
- You do not have to enter a deny everything statement at the end of your ACL; it is implicit.
- You can enter ACL entries in any order without any performance impact.
- For every eight TCAM entries, the ML-Series card uses one entry for TCAM management purposes.
- Do not set up conditions that result in packets getting lost. This situation can happen when a device or interface is configured to advertise services on a network that has ACLs that deny these packets.
- IP ACLs are not supported for double-tagged (QinQ) packets. They will, however, be applied to IP packets entering on a QinQ access port.

Creating IP ACLs

The following sections describe how to create numbered standard, extended, and named standard IP ACLs:

Creating Numbered Standard and Extended IP ACLs

Table 13-1 lists the global configuration commands used to create numbered standard and extended IP ACLs.

Table 13-1 Commands for Numbered Standard and Extended IP ACLs

Command	Purpose
ML_Series(config)# **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Defines a standard IP ACL using a source address and wildcard.
ML_Series(config)# **access-list** *access-list-number* {**deny** | **permit**} **any**	Defines a standard IP ACL using an abbreviation for the source and source mask of 0.0.0.0 255.255.255.255.
ML_Series(config)# **access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*] [**tos** *tos*]	Defines an extended IP ACL number and the access conditions.
ML_Series(config)# **access-list** *access-list-number* {**deny** | **permit**} *protocol* **any any**	Defines an extended IP ACL using an abbreviation for a source and source wildcard of 0.0.0.0 255.255.255.255, and an abbreviation for a destination and destination wildcard of 0.0.0.0 255.255.255.255.
ML_Series(config)# **access-list** *access-list-number* {**deny** | **permit**} *protocol* **host** *source* **host** *destination*	Defines an extended IP ACL using an abbreviation for a source and source wildcard of source 0.0.0.0, and an abbreviation for a destination and destination wildcard of destination 0.0.0.0.

Creating Named Standard IP ACLs

To create a named standard IP ACL, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	`ML_Series(config)# ip access-list standard name`	Defines a standard IP ACL using an alphabetic name.
Step 2	`ML_Series(config-std-nac1)# {deny | permit} {source [source-wildcard] | any}`	In access-list configuration mode, specifies one or more conditions as permitted or denied. This determines whether the packet is passed or dropped.
Step 3	`ML_Series(config)# exit`	Exits access-list configuration mode.

Creating Named Extended IP ACLs (Control Plane Only)

To create a named extended IP ACL, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	`ML_Series(config)# ip access-list extended name`	Defines an extended IP ACL using an alphabetic name.

Step 2	`ML_Series(config-ext-nacl)# {deny | permit} protocol source source-wildcard destination destination-wildcard [precedence precedence] [tos tos]`
	
	`or`
	
	`{deny | permit} protocol any any`
	
	`or`
	
	`{deny | permit} protocol host source host destination`
		In access-list configuration mode, specifies the conditions allowed or denied.
		
		Or:
		Defines an extended IP ACL using an abbreviation for a source and source wildcard of 0.0.0.0 255.255.255.255, and an abbreviation for a destination and destination wildcard of 0.0.0.0 255.255.255.255.
		
		Or:
		Defines an extended IP ACL using an abbreviation for a source and source wildcard of *source* 0.0.0.0, and an abbreviation for a destination and destination wildcard of *destination* 0.0.0.0.

Applying the ACL to an Interface

After you create an ACL, you can apply it to one or more interfaces. ACLs can be applied on either the inbound or the outbound direction of an interface. When controlling access to an interface, you can use a name or number. If a standard ACL is applied, the ML-Series card compares the source IP address with the ACL. To apply an ACL to one or more interfaces, use the command in Table 13-2.

Note IP standard ACLs applied to the ingress of a Bridge Group Virtual Interface (BVI) will be applied to all bridged IP traffic in the associated bridge-group, in addition to the BVI ingress traffic.

Table 13-2 Applying ACL to Interface

Command	Purpose
ip access-group {*access-list-number* | *name*} {**in** | **out**}	Controls access to an interface.

Modifying ACL TCAM Size

You can change the TCAM size by entering the **sdm access-list** command. For more information on ACL TCAM sizes, see the "Configuring Access Control List Size in TCAM" section on page 12-3. Example 13-1 provides an example of modifying and verifying ACLs.

Note To increase the ACL TCAM size, you must decrease another region's TCAM size, such as IP, IP multicast, or L2 switching.

Caution You need to increase the TCAM size if you see the following error message:

```
Warning:Programming TCAM entries failed
Please remove last ACL command to re-activate ACL operation.
!<ACL number or name> <IP or IPX> <INPUT_ACL or OUTPUT_ACL> from TCAM group for !<interface>
Please see the documentation to see if TCAM space can be
increased on this platform to alleviate the problem.
```

Example 13-1 Monitor and Verify ACLs

```
ML_Series# show ip access-lists 1
Standard IP access list 1
    permit 192.168.1.1
    permit 192.168.1.2
```

CHAPTER 14

Configuring Resilient Packet Ring on the ML-Series Card

Note The terms "Unidirectional Path Switched Ring" and "UPSR" may appear in Cisco literature. These terms do not refer to using Cisco ONS 15xxx products in a unidirectional path switched ring configuration. Rather, these terms, as well as "Path Protected Mesh Network" and "PPMN," refer generally to Cisco's path protection feature, which may be used in any topological network configuration. Cisco does not recommend using its path protection feature in any particular topological network configuration.

This chapter describes how to configure resilient packet ring (RPR) for the ML-Series card.

This chapter contains the following major sections:

Understanding RPR

RPR is a new MAC protocol operating at the Layer 2 level. It is well suited for transporting Ethernet over a SONET ring topology and enables multiple ML-Series cards to become one functional network segment or shared packet ring (SPR). RPR overcomes the limitations of earlier schemes, such as IEEE 802.1D Spanning Tree Protocol (STP), IEEE 802.1W Rapid Spanning Tree Protocol (RSTP), and

SONET, when used in this role. Although the IEEE 802.17 draft was used as reference for the Cisco ML-Series RPR implementation, the current ML-Series card RPR protocol does not comply with all clauses of IEEE 802.17.

Role of SONET Circuits

The ML-Series cards in an SPR must connect directly or indirectly through point-to-point STS circuits. The point-to-point STS circuits are configured on the ONS node and are transported over the ONS node's SONET topology with either protected or unprotected circuits.

On circuits unprotected by the SONET mechanism, RPR provides resiliency without using the capacity of the redundant protection path that a SONET protected circuit would require. This frees this capacity for additional traffic. RPR also utilizes the bandwidth of the entire ring and does not block segments like STP or RSTP.

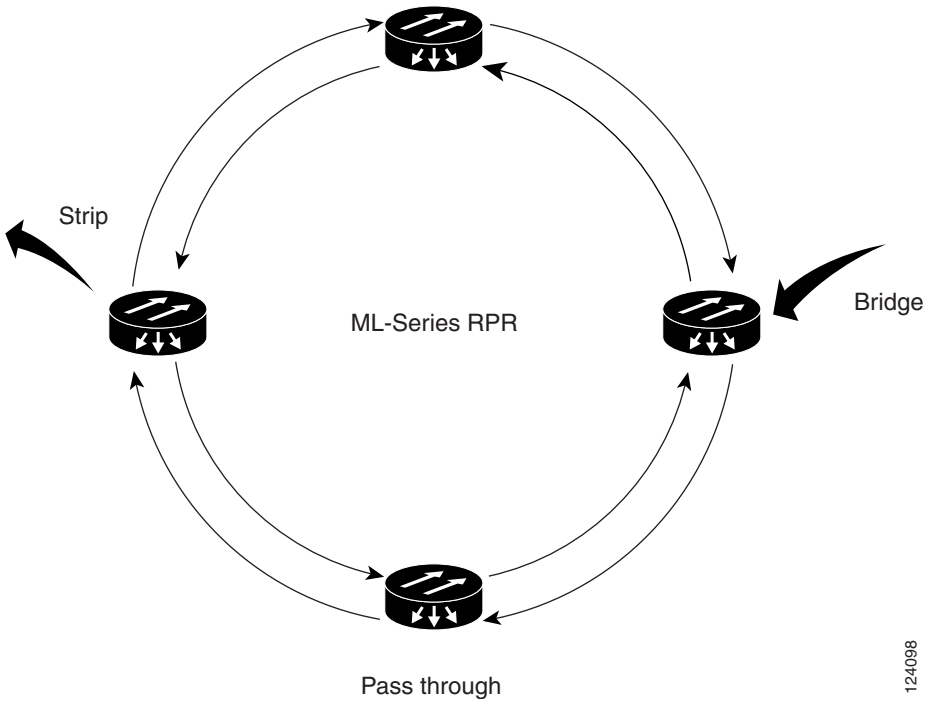
Packet Handling Operations

When an ML-Series card is configured with RPR and is made part of an SPR, the ML-Series card assumes a ring topology. If a packet is not destined for network devices bridged through the Ethernet ports of a specific ML-Series card, the ML-Series card simply continues to forward this transit traffic along the SONET circuit, relying on the circular path of the ring architecture to guarantee that the packet will eventually arrive at the destination. This eliminates the need to queue and process the packet flowing through the nondestination ML-Series card. From a Layer 2 or Layer 3 perspective, the entire RPR looks like one shared network segment.

An ML-Series card configured with RPR has three basic packet-handling operations: bridge, pass-through, and strip. Figure 14-1 illustrates these operations. Bridging connects and passes packets between the Ethernet ports on the ML-Series and the packet-over-SONET (POS) ports used for the SONET circuit circling the ring. Pass-through lets the packets continue through the ML-Series card and along the ring, and stripping takes the packet off the ring and discards it.

The RPR protocol, using the transmitted packet's header information, allows the interfaces to quickly determine the operation that needs to be applied to the packet. It also uses both the source and destination addresses of a packet to choose a ring direction. Flow-based load sharing helps ensure that all packets populated with equal source- and destination-address pairs will be sent in the same direction, and arrive at their destination in the correct order. Ring direction also enables the use of spatial reuse to increase overall ring aggregate bandwidth. Unicast packets are destination stripped. Destination stripping provides the ability to have simultaneous flows of traffic between different parts of an RPR. Traffic can be concurrently transmitted bidirectionally between adjacent nodes. It can also can span multiple nodes, effectively reusing the same ring bandwidth. Multicast packets are source stripped.

Figure 14-1 RPR Packet Handling Operations



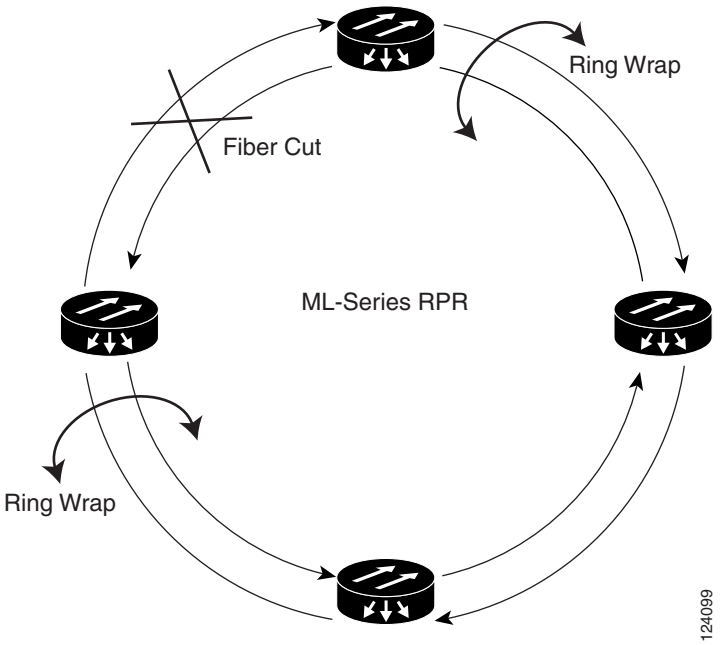
Ring Wrapping

RPR initiates ring wraps in the event of a fiber cut, node failure, node restoration, new node insertion, or other traffic problem. This protection mechanism redirects traffic to the original destination by sending it in the opposite direction around the ring after a link state change or after receiving SONET path level alarms. Ring wrapping on the ML-Series card allows convergence times of less than 50 ms. RPR convergence times are comparable to SONET and much faster than STP or RSTP.

RPR on the ML-Series card survives both unidirectional and bidirectional transmission failures within the ring. Unlike STP or RSTP, RPR restoration is scalable. Increasing the number of ML-Series cards in a ring does not increase the convergence time.

Ring wraps occur within 50 msec after the failure condition with the default **spr wrap immediate** configured. If **spr wrap delay** is configured, the wrap is delayed until the POS interface goes link-down. The link goes down after the time specified with the CLI **pos trigger delay** <msec>. If the circuits are VCAT then the Cisco IOS CLI command **pos vcat defect delayed** also needs to be configured. The delay helps ensure that when RPR is configured with SONET bandwidth protection, this Layer 1 protection has a chance to take effect before the Layer 2 RPR protection. If the interface goes down without a SONET error, then the carrier delay also take effect. Figure 14-2 illustrates ring wrapping.

Figure 14-2 RPR Ring Wrapping



In case of a ring failure, the ML-Series cards connected to the failed section of the RPR detect the failure through the SONET path alarms. When any ML-Series card receives this path-AIS signal, it wraps the POS interface that received the signal.

Note If the POS interfaces on the ML100T-8 cards on either 15310MA or 15310CL receives the SF-P condition, then the SPR ring does not wrap.

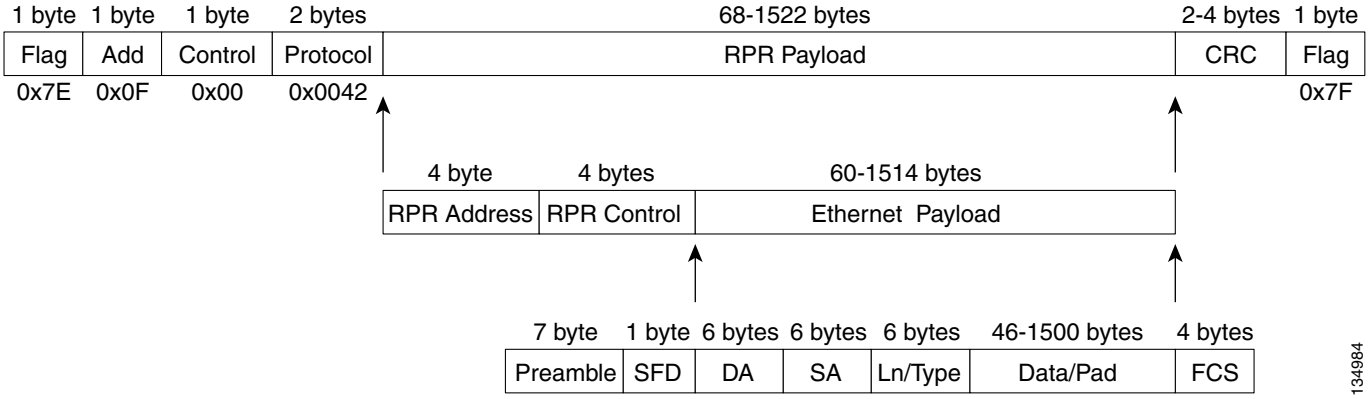
Note If the carrier delay time is changed from the default, the new carrier delay time must be configured on all the ML-Series card interfaces.

Note ML-Series card POS interfaces normally send an alarm for signal label mismatch failure in the STS path overhead (PDI-P) to the far end when the POS link goes down or when RPR wraps. ML-Series card POS interfaces do not send PDI-P to the far-end when PDI-P is detected, when a remote defection indication alarm (RDI-P) is being sent to the far end, or when the only defects detected are generic framing procedure (GFP)-loss of frame delineation (LFD), GFP client signal fail (CSF), virtual concatenation (VCAT)-loss of multiframe (LOM), or VCAT-loss of sequence (SQM).

RPR Framing Process

The ML-Series card uses a proprietary RPR frame and HDLC or GFP-F framing. It attaches the RPR frame header to each Ethernet frame and encapsulates the RPR frame into the SONET payload for transport over the SONET topology. The RPR header is removed at the egress ML-Series card. Figure 14-3 illustrates the RPR frame.

Figure 14-3 RPR Frame for ML-Series Card



The RPR framing and header includes a number of fields, including four bytes for source and destination station information and another four bytes for RPR control and quality of service (QoS). Figure 14-4 illustrates the RPR frame format. Table 14-1 defines the most important fields.

Figure 14-4 RPR Frame Fields

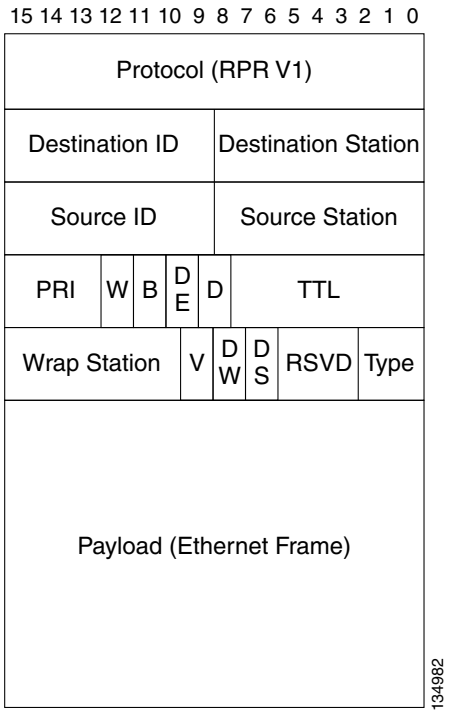


Table 14-1 Definitions of RPR Frame Fields

Destination Station	An eight-bit field specifying the MAC address of a specific ML-Series card in the RPR as the destination. It has two well-known addresses, 0xff for Multicast DA-MAC and 0x00 for Unknown DA-MAC.
Source Station	An eight-bit field specifying the MAC address of a specific ML-Series card in the RPR as the source.

Table 14-1 Definitions of RPR Frame Fields

PRI	A three-bit QoS class of service (CoS) field that establishes RPR priority.
DE	A one-bit field for the discard eligible flag.
TTL	A nine-bit field for the frame's time to live.
Type	A field indicating whether the packet is data or control.

MAC Address and VLAN Support

RPR increases the total number of MAC addresses supported because the MAC IDs of packets that pass through an ML-Series card are not recorded by that ML-Series card. The ML-Series card only records the MAC IDs of the packets that are bridged or stripped by that ML-Series card. This allows a greater number of MAC addresses in the collective address tables of the RPR.

VLANs on RPR require less interface configuration than VLANs on STP and RSTP, which require configuration on all the POS interfaces in the ring. RPR VLANs only require configuration on SPR interfaces that bridge or strip packets for that VLAN.

The ML-Series card still has an architectural maximum limit of 255 VLAN/bridge-group per ML-Series card. But because the ML-Series card only needs to maintain the MAC address of directly connected devices, a greater total number of connected devices are allowed on an RPR network.

RPR QoS

The ML-Series card's RPR relies on the QoS features of the ML-Series card for efficient bandwidth utilization with service level agreement (SLA) support. ML-Series card QoS mechanisms apply to all SONET traffic on the ML-Series card, whether passed-through, bridged, or stripped. For detailed RPR QoS information see the QoS on RPR section of Chapter 14, "Configuring Resilient Packet Ring on the ML-Series Card."

CTM and RPR

The Cisco Transport Manager (CTM) is an element management system (EMS) designed to integrate into an overall network management system (NMS) and interface with other higher level management tools. CTM supports RPR provisioning on ML-Series cards. For more information, refer to the *Cisco Transport Manager User Guide*.

Configuring RPR

You need to use both CTC and Cisco IOS to configure RPR for the ML-Series card. CTC is the graphical user interface (GUI) that serves as the enhanced craft tool for specific ONS node operations, including the provisioning of the point-to-point SONET circuits required for RPR. Cisco IOS is used to configure RPR on the ML-Series card and its interfaces.

Successfully creating an RPR requires several consecutive procedures:

1. Connecting the ML-Series Cards with Point-to-Point STS Circuits, page 14-7 (CTC or TL1)
2. Configuring CTC Circuits for RPR, page 14-7 (CTC or TL1)

3. Configuring RPR Characteristics and the SPR Interface on the ML-Series Card, page 14-9 (Cisco IOS)
4. Assigning the ML-Series Card POS Ports to the SPR Interface, page 14-11 (Cisco IOS)
5. Creating the Bridge Group and Assigning the Ethernet and SPR Interfaces, page 14-13 (Cisco IOS)
6. Verifying Ethernet Connectivity Between RPR Ethernet Access Ports, page 14-15 (Cisco IOS)
7. CRC Threshold Configuration and Detection, page 14-15

Note Transaction Language One (TL1) can be used to provision the required SONET point-to-point circuits instead of CTC.

Connecting the ML-Series Cards with Point-to-Point STS Circuits

You connect the ML-Series cards in an RPR through point-to-point STS circuits. These circuits use the SONET network and are provisioned using CTC in the normal manner for provisioning optical circuits.

Configuring CTC Circuits for RPR

These are the guidelines for configuring the CTC circuits required by RPR:

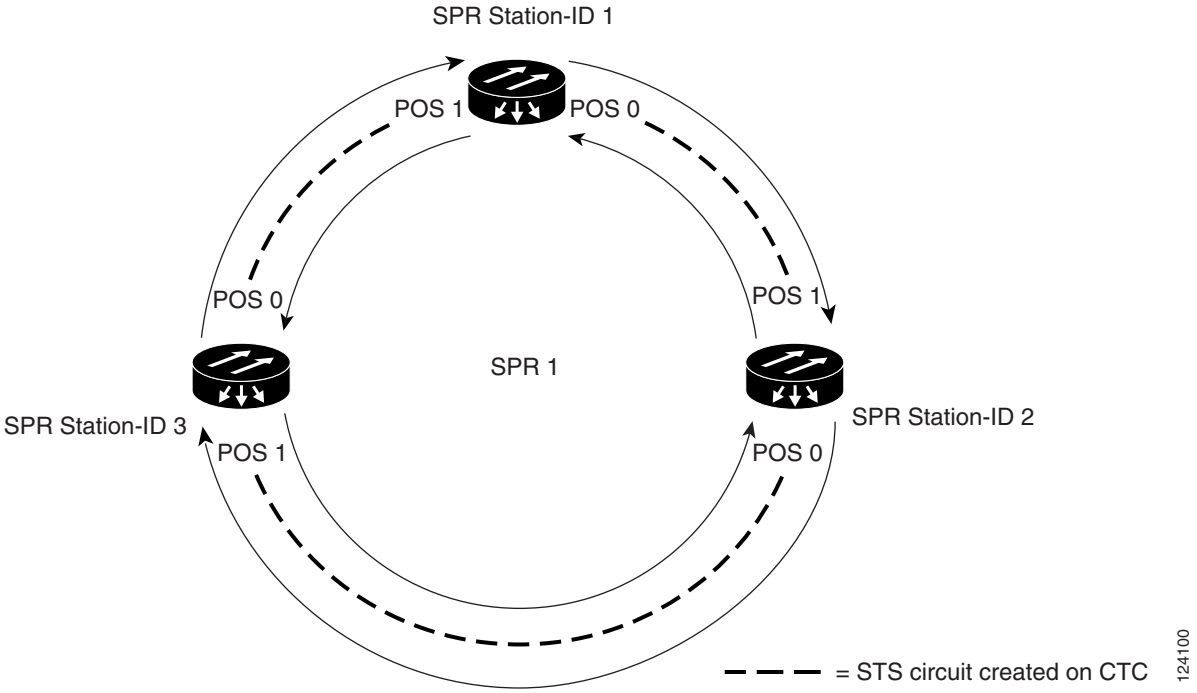
- Leave all CTC Circuit Creation Wizard options at their default settings, except **Fully Protected Path** in the Circuit Routing Preferences dialog box. **Fully Protected Path** provides SONET protection and should be unchecked. RPR normally provides the Layer 2 protection for SPR circuits.
- Check **Using Required Nodes and Spans** to route automatically in the Circuit Routing Preferences dialog box. If the source and destination nodes are adjacent on the ring, exclude all nodes except the source and destination in the Circuit Routing Preferences dialog box. This forces the circuit to be routed directly between source and destination and preserves STS circuits, which would be consumed if the circuit routed through other nodes in the ring. If there is a node or nodes that do not contain an ML-Series card between the two nodes containing ML-Series cards, include this node or nodes in the included nodes area in the Circuit Routing Preference dialog box, along with the source and destination nodes.
- Keep in mind that ML-Series card STS circuits do not support unrelated circuit creation options, such as the following check box titles in CTC, unidirectional traffic, creating cross-connects only (TL1-like), interdomain (unified control plane [UCP]), protected drops, subnetwork connection protection (SCNP), or path protectionpath selectors.
- A best practice is to configure SONET circuits in an east-to-west or west-to-east configuration, from Port 0 (east) to Port 1 (west) or Port 1 (east) to Port 0 (west), around the SONET ring. Do not configure Port 0 to Port 0 or Port 1 to Port 1. The east-to-west or west-to-east setup is also required in order for the CTM network management software to recognize the ML-Series configuration as an SPR.

Detailed CTC circuit procedures are available in the “Create Circuits and VT Tunnels” chapter of the *Cisco ONS 15454 Procedure Guide*.

CTC Circuit Configuration Example for RPR

Figure 14-5 illustrates an example of a three-node RPR.

Figure 14-5 Three-Node RPR Example



The three-node RPR in Figure 14-5 is used for all of the examples in the consecutive RPR procedures. Combining the examples will give you an end-to-end example of creating an RPR. It is assumed that the SONET node and its network is already active.

Caution The specific steps in the following procedure are for the topology shown in the example. Your own specific steps will vary according to your network. Do not attempt this procedure without obtaining a detailed plan or method of procedure from an experienced network architect.

To configure the circuits, you need to create three circuits in CTC:

- Create a circuit from Node 1, POS Port 0 to Node 2, POS Port 1.
- Create a circuit from Node 2, POS Port 0 to Node 3, POS Port 1.
- Create a circuit from Node 3, POS Port 0 to Node 1, POS Port 1.

Step 1 In CTC, log into Node 1 and navigate to the CTC card view for the ML-Series card that will be in the RPR.

Step 2 Click the **Circuits** > **Create** tabs.

The first page of the Circuit Creation wizard appears.

Step 3 In the Circuit Type list, select **STS**.

Step 4 Click **Next**.

The Circuit Attributes page appears.

Step 5 Type a circuit name in the Name field.

Step 6 Select the relevant size of the circuit from the Size drop-down list, and the appropriate state from the State list.

Step 7 Click **Next**.

The Source page appears.

Step 8 Select Node 1 as the source node from the node drop-down list.

Step 9 Select the ML-Series card from the Slot drop-down list, and choose 0 (POS) from the Port drop-down list.

Step 10 Click Next.

The Destination page appears.

Step 11 Select Node 2 as the destination node from the Node drop-down list.

Step 12 Select the ML-Series card from the Slot drop-down list, and choose 1 (POS) from the Port drop-down list.

Step 13 Click **Next**.

The Circuit Routing Preferences page appears.

Step 14 Uncheck the Fully Protected Path check box.

Step 15 Click **Next**.

The Circuit Constraints for Automatic Routing page appears.

Step 16 Click the Node 1 icon to select it and click **Next**.

The Route Review/Edit page appears.

Step 17 Click **Finish**.

You have now completed the initial circuit for the RPR.

Note A TPTFAIL alarm might appear on CTC when the circuit is created. This alarm will disappear after the POS ports are enabled during the "Assigning the ML-Series Card POS Ports to the SPR Interface" procedure on page 14-11.

Step 18 Build the second circuit between POS 0 on Node 2 and POS 1 on Node 3. Use the same procedure described in Steps 1 through 17, but substitute Node 2 for Node 1 and Node 3 for Node 2.

Step 19 Build the third circuit between POS 0 on Node 3 and POS 1 on Node 1. Use the same procedure described in Steps 1 through 17, but substitute Node 3 for Node 1 and Node 1 for Node 2.

Now all of the POS ports in all three nodes are connected by STS point-to-point circuits in an east-to-west pattern, as shown in Figure 14-5 on page 14-8.

Step 20 The CTC circuit process is complete.

Configuring RPR Characteristics and the SPR Interface on the ML-Series Card

You configure RPR on the ML-Series cards by creating an SPR interface using the Cisco IOS command-line interface (CLI). The SPR interface is a virtual interface for the SPR. An ML-Series card supports a single SPR interface with a single MAC address. It provides all the normal attributes of a Cisco IOS virtual interface, such as support for default routes.

An SPR interface is configured similarly to a EtherChannel (port-channel) interface. Instead of using the **channel-group** command to define the members, you use the **spr-intf-id** command. Like the port-channel interface, you configure the virtual SPR interface instead of the physical POS interface. An SPR interface is considered a trunk port, and like all trunk ports, subinterfaces must be configured for the SPR interface for it to join a bridge group.

The physical POS interfaces on the ML-Series card are the only members eligible for the SPR interface. One POS port is associated with the SONET circuit heading east around the ring from the node, and the other POS port is associated with the circuit heading west. When the SPR interface is used and the POS ports are associated, RPR encapsulation is used on the SONET payload.

Caution In configuring an SPR, if one ML-Series card is not configured with an SPR interface, but valid STS circuits connect this ML-Series card to the other ML-Series cards in the SPR, no traffic will flow between the properly configured ML-Series cards in the SPR, and no alarms will indicate this condition. Cisco recommends that you configure all of the ML-Series cards in an SPR before sending traffic.

Caution Do not use native VLANs for carrying traffic with RPR.

Note RPR on the ML-Series card is only supported with the default LEX encapsulation, a special CISCO-EOS-LEX encapsulation for use with Cisco ONS Ethernet line cards.

RPR needs to be provisioned on each ML-Series card that is in the RPR. To provision RPR, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# bridge irb`	Enables the Cisco IOS software to both route and bridge a given protocol on separate interfaces within a single ML-Series card.
Step 2	`Router(config)# interface spr 1`	Creates the SPR interface on the ML-Series card or enters the SPR interface configuration mode. The only valid SPR number is 1.
Step 3	`Router(config-if)# spr station-id` *station-ID-number*	Configures a station ID. The user must configure a different number for each SPR interface that attaches to the RPR. Valid station ID numbers range from 1 to 254.
Step 4	`Router(config-if)# spr wrap { immediate | delayed }`	(Optional) Sets the RPR ring wrap mode to either wrap traffic the instant it detects a SONET path alarm or to wrap traffic after the delay, which gives the SONET protection time to register the defect and declare the link down. Use **immediate** if RPR is running over unprotected SONET circuits. Use **delayed** for bidirectional line switched rings (BLSR), path protection, multiplex section-shared protection ring (MS-SPRing), or SNCP protected circuits. The default setting is **immediate**.

	Command	Purpose
Step 5	`Router(config-if)# carrier-delay msec` *milliseconds*	(Optional) Sets the carrier delay time. The default setting is 200 milliseconds, which is optimum for SONET protected circuits. **Note** If the carrier delay time is changed from the default, the new carrier delay time must be configured on all the ML-Series card interfaces.
Step 6	`Router(config-if)# [no] spr load-balance { auto | port-based }`	(Optional) Specifies the RPR load-balancing scheme for unicast packets. The **port-based** load balancing option maps even ports to the POS 0 interface and odd ports to the POS 1 interface. The default **auto** option balances the load based on the MAC addresses or source and destination addresses of the IP packet.
Step 7	`Router(config-if)# end`	Exits to privileged EXEC mode.
Step 8	`Router# copy running-config startup-config`	(Optional) Saves configuration changes to NVRAM.

Assigning the ML-Series Card POS Ports to the SPR Interface

Caution The SPR interface is the routed interface. Do not enable Layer 3 addresses or assign bridge groups on the POS interfaces assigned to the SPR interface.

Caution When traffic coming in on an SPR interface needs to be policed, the same input service policy needs to be applied to both POS ports that are part of the SPR interface.

The POS ports require LEX encapsulation to be used in RPR. The first step of RPR configuration is to set the encapsulation of POS 0 and POS 1 ports to LEX.

Each of the ML-Series card's two POS ports must also be assigned to the SPR interface. To configure LEX encapsulation and assign the POS interfaces on the ML-Series card to the SPR, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface pos 0`	Enters the interface configuration mode to configure the first POS interface that you want to assign to the SPR.
Step 2	`Router(config-if)# encapsulation lex`	Sets POS interface encapsulation as LEX (default). RPR on the ML-Series card requires LEX encapsulation.
Step 3 (	`Router(config-if)# spr-intf-id` *shared-packet-ring-number*	Assigns the POS interface to the SPR interface. The shared packet ring number must be 1, which is the only shared packet ring number that you can assign to the SPR interface.

	Command	Purpose
Step 4	`Router(config-if)# carrier-delay msec milliseconds`	(Optional) Sets the carrier delay time. The default setting is 200 msec, which is optimum for SONET protected circuits. **Note** The default unit of time for setting the carrier delay is seconds. The **msec** command resets the time unit to milliseconds.
Step 5	`Router(config-if)# pos trigger defect ber_sd-b3`	(Optional) Configures a trigger to bring down the POS interface when the SONET bit error rate exceeds the threshold set for the signal degrade alarm. Bringing the POS interface down initiates the RPR wrap. This command is recommended for all RPR POS interfaces, since excessive SONET bit errors can cause packet loss on RPR traffic. **Note** This command should not be used when a Cisco ONS 15310 is part of the ring. It may cause inconsistent RPR wrapping.
Step 6	`Router(config-if)# no shutdown`	Enables the POS port.
Step 7	`Router(config-if)# interface pos 1`	Enters the interface configuration mode to configure the second POS interface that you want to assign to the SPR.
Step 8	`Router(config-if)# encapsulation lex`	Sets POS interface encapsulation as LEX (default). RPR on the ML-Series card requires LEX encapsulation.
Step 9	`Router(config-if)# spr-intf-id shared-packet-ring-number`	Assigns the POS interface to the SPR interface. The shared packet ring number must be 1 (the same shared packet ring number that you assigned in Step 3), which is the only shared packet ring number that you can assign to the SPR interface.
Step 10	`Router(config-if)# carrier-delay msec milliseconds`	(Optional) Sets the carrier delay time. The default setting is 200 milliseconds, which is optimum for SONET protected circuits.
Step 11	`Router(config-if)# pos trigger defect ber_sd-b3`	(Optional) Configures a trigger to bring down the POS interface when the SONET bit error rate exceeds the threshold set for the signal degrade alarm. Bringing the POS interface down initiates the RPR wrap. This command is recommended for all RPR POS interfaces since excessive SONET bit errors can cause packet loss on RPR traffic.
Step 12	`Router(config-if)# no shutdown`	Enables the POS port.
Step 13	`Router(config-if)# end`	Exits to privileged EXEC mode.
Step 14	`Router# copy running-config startup-config`	(Optional) Saves the configuration changes to NVRAM.

Creating the Bridge Group and Assigning the Ethernet and SPR Interfaces

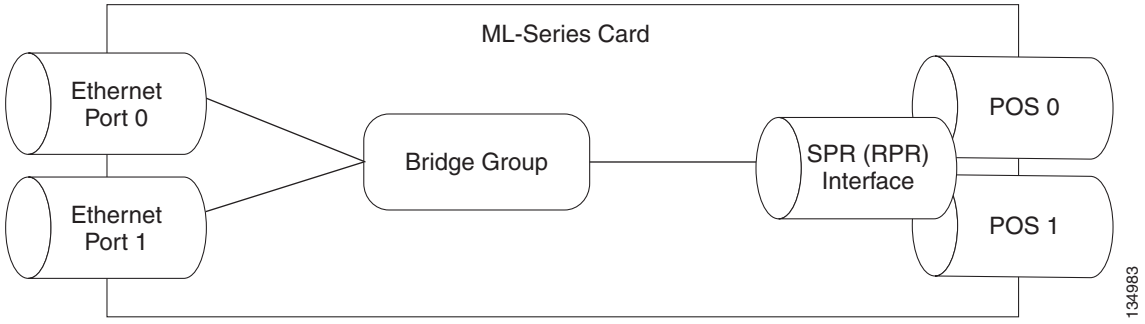
The default behavior of the ML-Series cards is that no traffic is bridged over the RPR even with the interfaces enabled. This is in contrast to many Layer 2 switches, including the Cisco Catalyst 6500 and the Cisco Catalyst 7600, which forward VLAN 1 by default. The ML-Series card will not forward any traffic by default, including untagged or VLAN 1 tagged packets.

For any RPR traffic to be bridged on an ML-Series card, a bridge group needs to be created for that traffic. Bridge groups maintain the bridging and forwarding between the interfaces on the ML-Series card and are locally significant. Interfaces not participating in a bridge group cannot forward bridged traffic.

To create a bridge group for RPR, you determine which Ethernet interfaces need to be in the same bridge group, create the bridge group, and associate these interfaces with the bridge group. Then associate the SPR interface with the same bridge group to provide transport across the RPR infrastructure.

Figure 14-6 illustrates a bridge group spanning the ML-Series card interfaces, including the SPR virtual interface of RPR.

Figure 14-6 RPR Bridge Group



Caution All Layer 2 network redundant links (loops) in the connecting network, except the RPR topology, must be removed for correct RPR operation. Or if loops exist, you must configure STP/RSTP.

To configure the needed interfaces, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface type number`	Enters interface configuration mode for the Ethernet interface joining the bridge group.
Step 2	`Router(config-if)# no shutdown`	Enables the interface.
Step 3	`Router(config-if)# bridge-group bridge-group-number`	Creates the specified bridge group and assigns the bridge group to the interface. Creating the bridge from the interface configuration disables STP or RSTP (spanning-disabled), which is recommended for RPR.
Step 4	`Router(config)# interface spr1`	Enters interface configuration mode for the SPR
Step 5	`Router(config-subif)# bridge-group bridge-group-number`	Associates the SPR interface to the specified bridge group.

RPR Cisco IOS Configuration Example

Figure 14-5 on page 14-8 shows a complete example of an RPR Cisco IOS configuration. The associated Cisco IOS code is provided in Examples 14-1, 14-2, and 14-3. The configuration assumes that ML-Series card POS ports are already linked by point-to-point SONET circuits configured through CTC.

Example 14-1 SPR Station-ID 1 Configuration

```
bridge irb

interface SPR1
no ip address
no keepalive
spr station-id 1
bridge-group 10
bridge-group 10 spanning-disabled
hold-queue 150 in

interface FastEthernet0
no ip address
bridge-group 10
bridge-group 10 spanning-disabled

interface FastEthernet1
no ip address
shutdown

interface POS0
no ip address
carrier-delay msec 0
spr-intf-id 1
crc 32

interface POS1
no ip address
carrier-delay msec 0
spr-intf-id 1
crc 32
!
```

Example 14-2 SPR Station-ID 2 Configuration

```
bridge irb

interface SPR1
no ip address
no keepalive
spr station-id 2
bridge-group 10
bridge-group 10 spanning-disabled

interface FastEthernet0
no ip address
bridge-group 10
bridge-group 10 spanning-disabled

interface FastEthernet1
no ip address
shutdown

interface POS0
```

```
no ip address
shutdown
spr-intf-id 1
crc 32

interface POS1
no ip address
spr-intf-id 1
crc 32
```

Example 14-3 SPR Station-ID 3 Configuration

```
bridge irb

interface SPR1
no ip address
no keepalive
spr station-id 3
bridge-group 10
bridge-group 10 spanning-disabled
hold-queue 150 in

interface FastEthernet0
no ip address
bridge-group 10
bridge-group 10 spanning-disabled

interface FastEthernet1
no ip address
shutdown

interface POS0
no ip address
spr-intf-id 1
crc 32

interface POS1
no ip address
spr-intf-id 1
crc 32
!
```

Verifying Ethernet Connectivity Between RPR Ethernet Access Ports

After successfully completing the procedures to provision an RPR, you can test Ethernet connectivity between the Ethernet access ports on the separate ML-Series cards using your standard Ethernet connectivity testing.

CRC Threshold Configuration and Detection

You can configure a span shutdown when the ML-Series card receives CRC errors at a rate that exceeds the configured threshold and configured soak time. For this functionality to work in an SPR ring, make the configurations on the POS members of SPR interface specified in CRC Threshold Configuration, page 4-11.

Monitoring and Verifying RPR

After RPR is configured, you can monitor its status using the **show interface spr 1 command** (Example 14-4) or the **show run interface spr 1** command (Example 14-5).

Example 14-4 Example of show interface spr 1 Output

```
ML-Series# show interfaces spr 1

SPR1 is up, line protocol is up
  Hardware is POS-SPR, address is 0005.9a39.77f8 (bia 0000.0000.0000)
  MTU 1500 bytes, BW 290304 Kbit, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
 Encapsulation: Cisco-EoS-LEX, loopback not set
  Keepalive not set
  DTR is pulsed for 27482 seconds on reset, Restart-Delay is 65 secs
  ARP type: ARPA, ARP Timeout 04:00:00
    No. of active members in this SPR interface: 2
        Member 0 : POS1
        Member 1 : POS0
  Last input 00:00:38, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/150/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/80 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     37385 packets input, 20993313 bytes
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
              0 parity
     2 input errors, 2 CRC, 0 frame, 0 overrun, 0 ignored
     0 input packets with dribble condition detected
     37454 packets output, 13183808 bytes, 0 underruns
     0 output errors, 0 applique, 4 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions
```

Example 14-5 Example of show run interface spr 1 Output

```
ML-Series# show run interface spr 1

Building configuration...
Current configuration : 141 bytes
interface SPR1
 no ip address
 no keepalive
 spr station-id 2
 bridge-group 10
 bridge-group 10 spanning-disabled
 hold-queue 150 in
end
```

Add an ML-Series Card into an RPR

An existing RPR might need an ML-Series card added. This can be done without taking down data traffic due to the RPR wrapping capability and ring architecture. You can add the ML-Series card in concert with the addition of the node containing the card into the underlying SONET architecture. You can also add an ML-Series card to a node that is already part of the SONET topology.

The following example has a two-node RPR with two STS circuits connecting the ML-Series cards. One circuit will be deleted. The RPR will wrap traffic on the remaining circuit with as little as a one ping loss. The third node and ML-Series card are then added in, and the spans and circuits for this card are created.

Figure 14-7 shows the existing two-node RPR with the single STS circuit and span that will be deleted.

Figure 14-7 Two-Node RPR Before the Addition

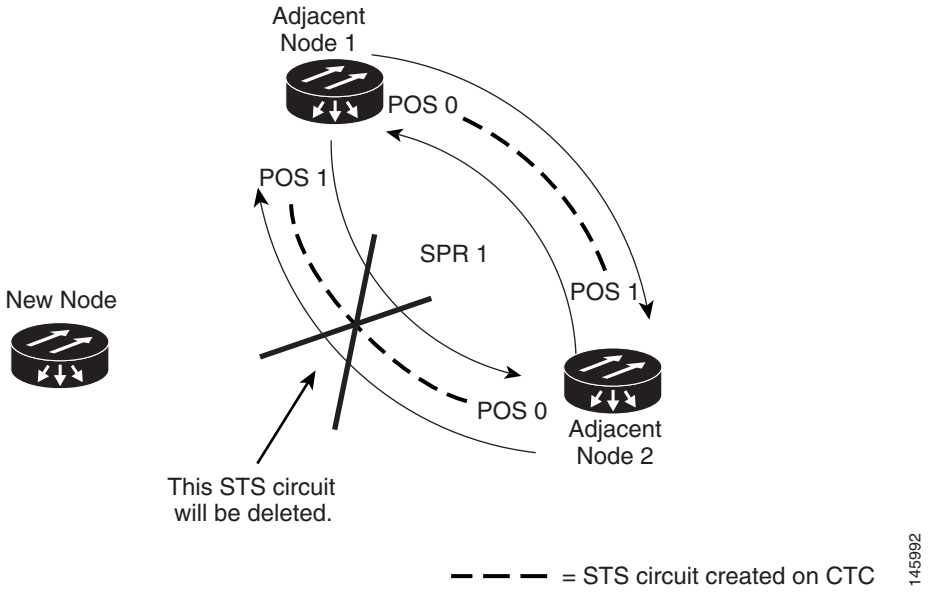
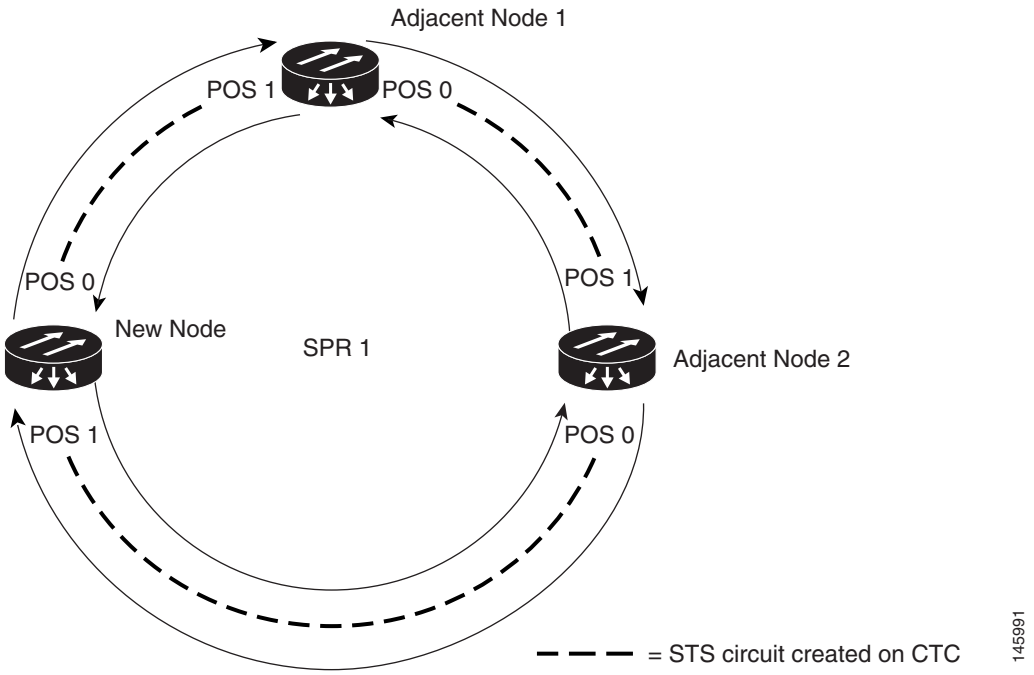


Figure 14-8 shows the RPR after the third node is added with the two new STS circuits and spans that will be added.

Figure 14-8 Three-Node RPR After the Addition



To add an ML-Series card to the RPR, you need to complete several general actions:

- Force away any existing non-ML-Series card circuits, such as DS-1, that use the span that will be deleted.
- Shut down the POS ports on the adjacent ML-Series cards for the STS circuit that will be deleted to initiate the RPR wrap.
- Test Ethernet connectivity between the access ports on the existing adjacent ML-Series cards with a test set to ensure that the RPR wrapped successfully.
- Delete the STS circuit that will be replaced by the new circuits. (In Figure 14-7, this is the circuit between Adjacent Node 2, POS 0 and Adjacent Node 1, POS 1.)
- Insert the new node into the ring topology if the node is not already part of the topology.
- Install the ML-Series card and load your initial configuration file or otherwise do an initial configuration of the ML-Series card.
- Ensure the new node is configured with RPR before its POS ports are manually enabled or enabled through the configuration file.
- Create an STS circuit from one of the POS ports of an existing adjacent ML-Series card to a POS port on the new ML-Series card. (In Figure 14-8, this is the circuit between Adjacent Node 2, POS Port 0 and New Node, POS Port 1.)
- Create a second STS circuit from one of the POS ports of the other existing adjacent ML-Series card to the remaining POS port on the new ML-Series card. (In Figure 14-8, this is the circuit between New Node, POS Port 0 and Adjacent Node 1, POS Port 1.)
- Configure the new ML-Series card to join the RPR and enable the POS ports, if the initial configuration file did not already do this.
- Enable the POS ports on the existing adjacent ML-Series cards that connect to the new ML-Series card. (In Figure 14-8, these are Adjacent Node 1, POS Port 1 and Adjacent Node 2, POS Port 0.)

- Test Ethernet connectivity between the access ports on the new ML-Series card with a test set to validate the newly created three-node RPR.
- Monitor Ethernet traffic and existing routing protocols for at least an hour after the node insertion.

Caution The specific steps in the following procedure are for the topology in the example. Your own steps will vary according to your network design. Do not attempt this procedure without obtaining a detailed plan or method of procedure from an experienced network architect.

Adding an ML-Series Card into an RPR

To add an ML-Series card to the RPR in the example, complete the following procedure:

Step 1 Start a Cisco IOS CLI session for the ML-Series card in the first adjacent node. This is Adjacent Node 1 in Figure 14-7.

Step 2 Complete the following Cisco IOS configuration on the ML-Series card in the first adjacent node, beginning in global configuration mode:

a.	`Router(config)# interface pos` ***interface-number***	Enters interface configuration mode for the POS port at one endpoint of the circuit to be deleted.
b.	`Router(config-if)# shutdown`	Closes the interface, which initiates the RPR wrap.

Step 3 Start a Cisco IOS CLI session for the ML-Series card in Adjacent Node 2, as shown in Figure 14-7.

Step 4 Complete the following Cisco IOS configuration on the Adjacent Node 2 ML-Series card, beginning in global configuration mode:

a.	`Router(config)# interface pos` ***interface-number***	Enters interface configuration mode for the POS port at one endpoint of the circuit to be deleted.
b.	`Router(config-if)# shutdown`	Closes the interface.

Step 5 In CTC, log into Adjacent Node 1.

Step 6 Double-click the ML-Series card in Adjacent Node 1.

The card view appears.

Step 7 Click the **Circuits** tab.

Step 8 Click the **Circuits** subtab.

Step 9 Identify the appropriate STS circuit by looking under the source column and destination column for the circuit entry that matches the POS ports at the endpoints of the circuit to be deleted.

The circuit entry is in *node-name/card-slot/port-number* format, such as Node-1/s12(ML100T)/pPOS-0.

Step 10 Click the circuit entry to highlight it.

Step 11 Click **Delete**.

A confirmation dialog box appears.

Step 12 Click **Yes**.

Step 13 Use a test set to verify that Ethernet connectivity still exists between the Ethernet access ports on Adjacent Node 1 and Adjacent Node 2.

Note The SPR interface and the Ethernet interfaces on the ML-Series card must be in a bridge group in order for RPR traffic to bridge the RPR.

Step 14 If the new node is not already an active node in the SONET ring topology, add the node to the ring. Refer to the "Add and Remove Nodes" chapter of the *Cisco ONS 15454 Procedure Guide*.

Step 15 If the ML-Series card in the new node is not already installed, install the card in the node. Refer to the "Install the Cisco ONS 15310-CL" or "Install the Cisco ONS 15310-MA" chapters of the *Cisco ONS 15454 Procedure Guide*.

Step 16 Upload the initial startup configuration file for the new ML-Series card. If you do not have a prepared startup configuration file, manually create a startup configuration file.

Caution Ensure the new node is configured with RPR before its POS ports are manually enabled or enabled through the configuration file.

Step 17 Build an STS circuit with a circuit state of In Service (IS) from the available POS port on Adjacent Node 1 to the New Node, as shown in Figure 14-8. On the New Node, use the POS port with the interface-number that does not match the interface-number of the available POS port on Adjacent Node 1. For example, POS Port 0 on Adjacent Node 1would connect to POS Port 1 on the New Node.

For detailed steps for building the circuit, see the "Configuring CTC Circuits for RPR" section on page 14-7.

Note A best practice is to configure SONET circuits in an east-to-west or west-to-east configuration, from Port 0 (east) to Port 1 (west) or Port 1 (east) to Port 0 (west), around the SONET ring.

Step 18 Build an STS circuit with a circuit state of IS from the available POS port on Adjacent Node 2 to the remaining POS port on the New Node, as shown in Figure 14-8.

Step 19 Start or resume a Cisco IOS CLI session for the ML-Series card in Adjacent Node 1, as shown in Figure 14-7.

Step 20 Complete the following Cisco IOS configuration, beginning in global configuration mode:

a.	`Router(config)# interface pos` *interface-number*	Enters interface configuration mode for the POS port at one endpoint of the first newly created circuit.
b.	`Router(config-if)# no shutdown`	Enables the port.

Step 21 Start a Cisco IOS CLI session for the ML-Series card in Adjacent Node 2, as shown in Figure 14-7.

Step 22 Complete the following Cisco IOS configuration on the Adjacent Node 2 ML-Series card, beginning in global configuration mode:

a.	`Router(config)# interface pos` *interface-number*	Enters interface configuration mode for the POS port at one endpoint of the second newly created circuit.
b.	`Router(config-if)# no shutdown`	Enables the port.

Step 23 Use a test set to verify that Ethernet connectivity exists on the RPR.

Step 24 Monitor Ethernet traffic and routing tables for at least one hour after the node insertion.

Stop. You have completed this procedure.

Delete an ML-Series Card from an RPR

An existing RPR might need an ML-Series card deleted. This can be done without taking down data traffic due to the RPR wrapping capability and ring architecture.

The following example has a three-node RPR with three STS circuits connecting the ML-Series cards. Two circuits will be deleted. The RPR will wrap traffic on the remaining circuit with as little as a one ping loss. The third node and ML-Series card are then deleted and a new STS circuit is created between the two remaining cards.

Figure 14-9 shows the existing three-node RPR with all three STS circuits and spans. Figure 14-10 shows the RPR after the third node, circuits, and spans are deleted and the new STS circuit and span are added.

Figure 14-9 Three-Node RPR Before the Deletion

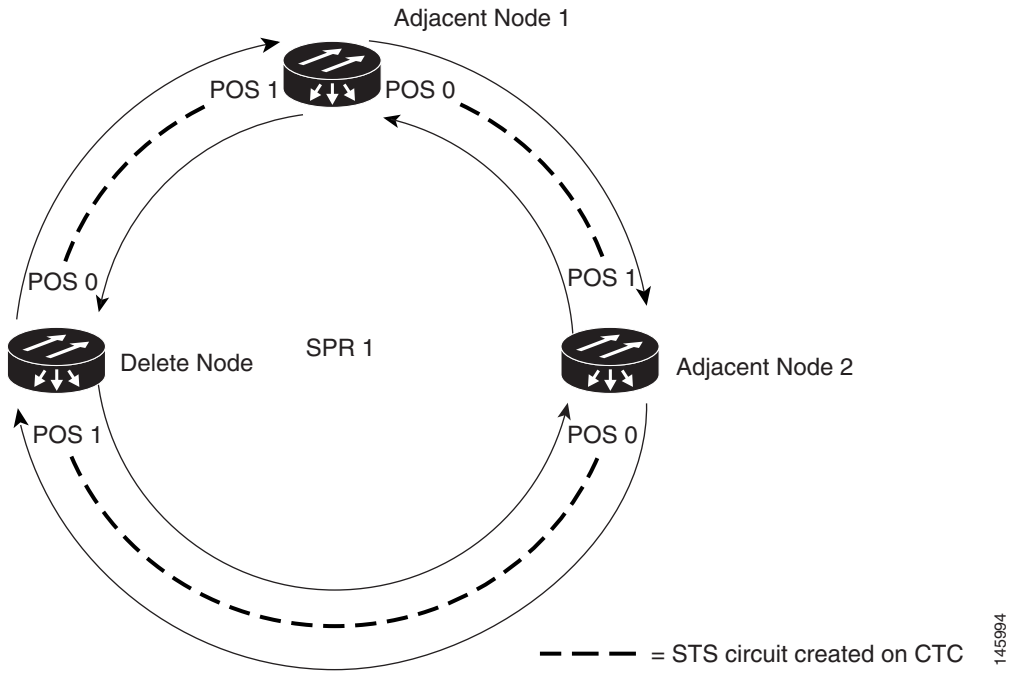
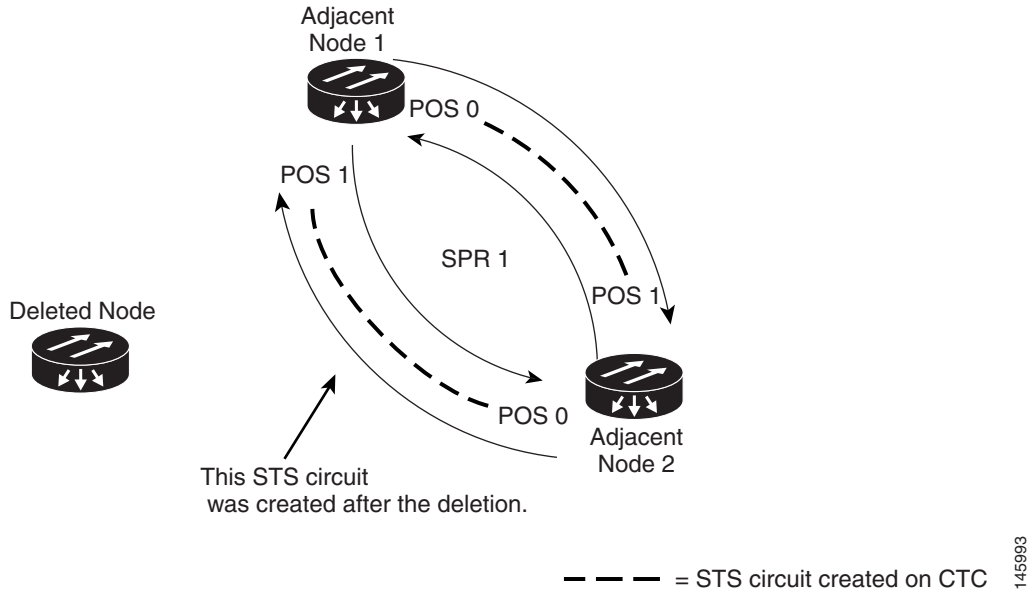


Figure 14-10 Two-Node RPR After the Deletion



To delete an ML-Series card from the RPR, you need to complete several general actions:

- Force away any existing non-ML-Series card circuits, such as DS-1, that use the spans that will be deleted.
- Shut down the POS ports on the adjacent ML-Series cards for the STS circuits that will be deleted to initiate the RPR wrap.

- Test Ethernet connectivity between the access ports on the existing adjacent ML-Series cards with a test set to ensure that the RPR wrapped successfully.
- Delete the two STS circuits that will be replaced by the new circuits. (In Figure 14-9, this is the circuit between the Delete Node and one Adjacent Node, and the circuit between the Delete Node and the other Adjacent Node.)
- Remove the Delete Node from the ring topology if desired.
- Physically remove the delete ML-Series card from the node if desired.
- Create an STS circuit from the available POS port of one of the remaining adjacent ML-Series cards to the available POS port on the other remaining adjacent ML-Series card. (In Figure 14-10, this is the circuit between Adjacent Node 2, POS Port 0 and Adjacent Node 1, POS Port 1.)
- Enable the POS ports on the existing adjacent ML-Series cards.(In Figure 14-10, this is the Adjacent Node 2, POS Port 0 and the Adjacent Node 1, POS Port 1.)
- Test Ethernet connectivity between the access ports on the adjacent ML-Series card with a test set to validate the two-node RPR.
- Monitor Ethernet traffic and existing routing protocols for at least an hour after the node deletion.

Caution The specific steps in the following procedure are for the topology in the example. Your own steps will vary according to your network design. Do not attempt this procedure without obtaining a detailed plan or method of procedure from an experienced network architect.

Deleting an ML-Series Card from an RPR

To delete an ML-Series card from an RPR, complete the following procedure:

Step 1 Start a Cisco IOS CLI session for the ML-Series card on the first adjacent node. This is Adjacent Node 1 in Figure 14-9.

Step 2 Complete the following Cisco IOS configuration on the ML-Series card in the first adjacent node, beginning in global configuration mode:

a.	`Router(config)# interface pos interface-number`	Enters interface configuration mode for the POS port at the end of the circuit directly connected to the Delete Node.
b.	`Router(config-if)# shutdown`	Closes the interface, which initiates the RPR wrap.

Step 3 Start a Cisco IOS CLI session for the ML-Series card in Adjacent Node 2, as shown in Figure 14-9.

Step 4 Complete the following Cisco IOS configuration on the Adjacent Node 2 ML-Series card, beginning in global configuration mode:

a.	`Router(config)# interface pos` *interface-number*	Enters interface configuration mode for the POS port at the end of the circuit directly connected to the Delete Node.
b.	`Router(config-if)# shutdown`	Closes the interface.

Step 5 Log into Adjacent Node 1 with CTC.

Step 6 Double-click the ML-Series card in Adjacent Node 1.

The card view appears.

Step 7 Click the **Circuits** tab.

Step 8 Click the **Circuits** subtab.

Step 9 Identify the appropriate STS circuit by looking under the source column and destination column for the circuit entry that matches the POS ports at the endpoints of the first circuit to be deleted.

The circuit entry is in *node-name/card-slot/port-number* format, such as Node-1/s12(ML100T)/pPOS-0.

Step 10 Click the circuit entry to highlight it.

Step 11 Click **Delete**.

A confirmation dialog box appears.

Step 12 Click **Yes**.

Step 13 Verify that Ethernet connectivity still exists between the Ethernet access ports on Adjacent Node 1 and Adjacent Node 2 by using a test set.

Note The SPR interface and the Ethernet interfaces on the ML-Series card must be in a bridge group in order for RPR traffic to bridge the RPR.

Step 14 Log into Adjacent Node 2 with CTC.

Step 15 Double-click the ML-Series card in Adjacent Node 2.

The card view appears.

Step 16 Click the **Circuits** tab.

Step 17 Click the **Circuits** subtab.

Step 18 Identify the appropriate STS circuit by looking under the source column and destination column for the circuit entry that matches the POS ports at the endpoints of the second circuit to be deleted.

The circuit entry is in *node-name/card-slot/port-number* format, such as Node-1/s12(ML100T)/pPOS-0.

Step 19 Click the circuit entry to highlight it.

Step 20 Click **Delete**.

The confirmation dialog box appears.

Step 21 Click **Yes**.

Step 22 If the new node will no longer be an active node in the SONET ring topology, delete the node from the ring. Refer to the "Add and Remove Nodes" chapter of the *Cisco ONS 15454 Procedure Guide*.

Step 23 If the ML-Series card in the new node is to be deleted in CTC and physically removed, do so now. Refer to the "Install the Cisco ONS 15310-CL" or "Install the Cisco ONS 15310-MA" chapters of the *Cisco ONS 15454 Procedure Guide*.

Step 24 Build an STS circuit with a circuit state of IS from the available POS port on Adjacent Node 1 to the available POS port on Adjacent Node 2, as shown in Figure 14-10. For detailed steps on building the circuit, see "Configuring CTC Circuits for RPR" section on page 14-7.

Note A best practice is to configure SONET circuits in an east-to-west or west-to-east configuration, from Port 0 (east) to Port 1 (west) or Port 1 (east) to Port 0 (west), around the SONET ring.

Step 25 Start or resume a Cisco IOS CLI session for the ML-Series card in Adjacent Node 1.

Step 26 Complete the following Cisco IOS configuration for the ML-Series card in Adjacent Node 1, beginning in global configuration mode:

a.	`Router(config)# interface pos` *interface-number*	Enters interface configuration mode for the POS port at one endpoint of the first newly created circuit.
b.	`Router(config-if)# no shutdown`	Enables the port.

Step 27 Start a Cisco IOS CLI session for the ML-Series card in Adjacent Node 2.

Step 28 Complete the following Cisco IOS configuration on the Adjacent Node 2 ML-Series card, beginning in global configuration mode:

a.	`Router(config)# interface pos` *interface-number*	Enters interface configuration mode for the POS port at one endpoint of the second newly created circuit.
b.	`Router(config-if)# no shutdown`	Enables the port.

Step 29 Use a test set to verify that Ethernet connectivity exists on the RPR.

Step 30 Monitor Ethernet traffic and routing tables for at least one hour after the node deletion.

Stop. You have completed this procedure.

Cisco Proprietary RPR KeepAlive

Please see *Cisco ONS 15454 Ether Guide Chapter 17*

Configuring Cisco Proprietary RPR KeepAlive

Please see *Cisco ONS 15454 Ether Guide Chapter 17*

Monitoring Cisco Propretary RPR KeepAlive

Please see *Cisco ONS 15454 Ether Guide Chapter 17*

Cisco Proprietary RPR Shortest Path

Please see *Cisco ONS 15454 Ether Guide Chapter 17*

Configuring Shortest Path and Topology Discovery

Please see*Cisco ONS 15454 Ether Guide Chapter 17*

Monitoring and Verifying Shortest Path andTopolgy Discovery

Please see*Cisco ONS 15454 Ether Guide Chapter 17*

Redundant Interconnect

Redundant Interconnect is only supported on 454 platforms

CHAPTER 15

Configuring Security for the ML-Series Card

This chapter describes the security features of the ML-Series card and includes the following major sections:

Understanding Security

The ML-Series card includes several security features. Some of these features operate independently from the ONS node where the ML-Series card is installed. Others are configured using the Cisco Transport Controller (CTC) or Transaction Language One (TL1).

Security features configured with Cisco IOS include:

- Cisco IOS login enhancements
- Secure Shell (SSH) connection
- authentication, authorization, and accounting/Remote Authentication Dial-In User Service (AAA/RADIUS) stand alone mode
- Cisco IOS basic password (For information on basic Cisco IOS password configuration, see the "Passwords" section on page 3-6.)

Security features configured with CTC or TL1 include:

- disabled console port
- AAA/RADIUS relay mode

Disabling the Console Port on the ML-Series Card

There are several ways to access the Cisco IOS running on the ML-Series card, including a direct connection to the console port, which is the RJ-11 serial port on the front of the card. Users can increase security by disabling this direct connection, which is enabled by default. This prevents console port input without preventing any console port output, such as Cisco IOS error messages.

You can disable console port access through CTC or TL1. To disable it with CTC, at the card-level view of the ML-Series card, click under the **IOS** tab and uncheck the **Enable Console Port Access** box and click **Apply.** The user must be logged in at the Superuser level to complete this task.

To disable it using TL1, refer to the *Cisco ONS SONET TL1 Command Guide*.

Secure Login on the ML-Series Card

The ML-Series card supports the Cisco IOS login enhancements integrated into Cisco IOS Release 12.2(25)S and introduced in Cisco IOS Release 12.3(4)T. The enhancements allow users to better secure the ML-Series card when creating a virtual connection, such as Telnet, Secure Shell, or HTTP. The secure login feature records successful and failed login attempts for vty sessions (audit trail) on the ML-Series card. These features are configured using the Cisco IOS command-line interface (CLI.)

For more information, including step-by-step configuration examples, refer to the Cisco IOS Release 12.2(25)S feature guide module *Cisco IOS Login Enhancements* at http://www.cisco.com/en/US/products/sw/iosswrel/ps1838/products_feature_guides_list.html.

Secure Shell on the ML-Series Card

This section describes how to configure the SSH feature and contains this information:

For other SSH configuration examples, see the "SSH Configuration Examples" section in the "Configuring Secure Shell" chapter of the *Cisco IOS Security Configuration Guide, Cisco IOS Release 12.2*, at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fothersf.htm

Note For complete syntax and usage information for the commands used in this section, see the command reference for Cisco IOS Release 12.2 at the URL: http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/index.htm.

Understanding SSH

The ML-Series card supports SSH, both version 1 (SSHv1) and version 2 (SSHv2). SSHv2 offers security improvements over SSHv1 and is the default choice on the ML-Series card.

SSH has two applications, an SSH server and SSH client. The ML-Series card only supports the SSH server and does not support the SSH client. The SSH server in Cisco IOS software works with publicly and commercially available SSH clients.

The SSH server enables a connection into the ML-Series card, similar to an inbound Telnet connection, but with stronger security. Before SSH, security was limited to the native security in Telnet. SSH improves on this by allowing the use of Cisco IOS software authentication.

The ONS node also supports SSH. When SSH is enabled on the ONS node, you use SSH to connect to the ML-Series card for Cisco IOS CLI sessions.

Note Telnet access to the ML-Series card is not automatically disabled when SSH is enabled. The user can disable Telnet access with the vty line configuration command **transport input ssh**.

Configuring SSH

This section has configuration information:

- Configuration Guidelines, page 15-3
- Setting Up the ML-Series Card to Run SSH, page 15-3 (required)
- Configuring the SSH Server, page 15-4 (required)

Configuration Guidelines

Follow these guidelines when configuring the ML-Series card as an SSH server:

- The new model of AAA and a AAA login method must be enabled. If not previously enabled, complete the "Configuring AAA Login Authentication" section on page 15-11.
- A Rivest, Shamir, and Adelman (RSA) key pair generated by a SSHv1 server can be used by an SSHv2 server, and the reverse.
- If you get CLI error messages after entering the **crypto key generate rsa** global configuration command, an RSA key pair has not been generated. Reconfigure the hostname and domain, and then enter the **crypto key generate rsa** command. For more information, see the "Setting Up the ML-Series Card to Run SSH" section on page 15-3.
- When generating the RSA key pair, the message `No host name specified` might appear. If it does, you must configure a hostname by using the **hostname** global configuration command.
- When generating the RSA key pair, the message `No domain specified` might appear. If it does, you must configure an IP domain name by using the **ip domain-name** global configuration command.

Setting Up the ML-Series Card to Run SSH

Follow these steps to set up your ML-Series card to run as an SSH server:

1. Configure a hostname and IP domain name for the ML-Series card.
2. Generate an RSA key pair for the ML-Series card, which automatically enables SSH.
3. Configure user authentication for local or remote access. This step is required.

Beginning in privileged EXEC mode, follow these steps to configure a hostname and an IP domain name and to generate an RSA key pair.

	Command	Purpose
Step 1	**Router #configure terminal**	Enter global configuration mode.
Step 2	**Router (config)# hostname** *hostname*	Configure a hostname for your ML-Series card.
Step 3	**Router (config)# ip domain-name** *domain_name*	Configure a host domain for your ML-Series card.
Step 4	**Router (config)# crypto key generate rsa**	Enable the SSH server for local and remote authentication on the ML-Series card and generate an RSA key pair. When you generate RSA keys, you are prompted to enter a modulus length. The default modulus length is 512 bits. A longer modulus length might be more secure, but it takes longer to generate and to use.
Step 5	**Router (config)# ip ssh timeout** *seconds*	Specify the timeout value in seconds; the default is 120 seconds. The range is 0 to 120 seconds. This parameter applies to the SSH negotiation phase. After the connection is established, the ML-Series card uses the default timeout values of the CLI-based sessions. By default, up to five simultaneous, encrypted SSH connections for multiple CLI-based sessions over the network are available (session 0 to session 4). After the execution shell starts, the CLI-based session timeout value returns to the default of 10 minutes.
Step 6	**Router (config)# ip ssh authentication-retries** *number*	Specify the number of times that a client can reauthenticate to the server. The default is 3; the range is 0 to 5.
Step 7	**Router (config)# end**	Return to privileged EXEC mode.
Step 8	**Router # show ip ssh** or **Router # show ssh**	Displays the version and configuration information for your SSH server. Displays the status of the SSH server on the ML-Series card.
Step 9	**Router #** show crypto key mypubkey rsa	Displays the generated RSA key pair associated with this ML-Series card.
Step 10	**Router # copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete the RSA key pair, use the **crypto key zeroize rsa** global configuration command. After the RSA key pair is deleted, the SSH server is automatically disabled.

Configuring the SSH Server

Beginning in privileged EXEC mode, follow these steps to configure the SSH server:

	Command	Purpose
Step 1	**Router # configure terminal**	Enter global configuration mode.
Step 2	**Router (config)# ip ssh version** [**1** | **2**]	(Optional) Configure the ML-Series card to run SSH Version 1 or SSH Version 2. • **1**—Configure the ML-Series card to run SSH Version 1. • **2**—Configure the ML-Series card to run SSH Version 2. If you do not enter this command or do not specify a keyword, the SSH server selects the latest SSH version supported by the SSH client. For example, if the SSH client supports SSHv1 and SSHv2, the SSH server selects SSHv2.
Step 3	**Router (config)# ip ssh timeout** *seconds*	Specify the timeout value in seconds; the default is 120 seconds. The range is 0 to 120 seconds. This parameter applies to the SSH negotiation phase. After the connection is established, the ML-Series card uses the default timeout values of the CLI-based sessions. By default, up to five simultaneous, encrypted SSH connections for multiple CLI-based sessions over the network are available (session 0 to session 4). After the execution shell starts, the CLI-based session timeout value returns to the default of 10 minutes.
Step 4	**Router (config)# ip ssh authentication-retries** *number*	Specify the number of times that a client can reauthenticate to the server. The default is 3; the range is 0 to 5.
Step 5	**Router (config)# end**	Return to privileged EXEC mode.
Step 6	**Router # show ip ssh** or **Router # show ssh**	Show the version and configuration information for your SSH server. Show the status of the SSH server connections on the ML-Series card.
Step 7	**Router # copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default SSH control parameters, use the **no ip ssh** {**timeout** | **authentication-retries**} global configuration command.

Displaying the SSH Configuration and Status

To display the SSH server configuration and status, use one or more of the privileged EXEC commands in Table 15-1.

Table 15-1 Commands for Displaying the SSH Server Configuration and Status

Command	Purpose
show ip ssh	Shows the version and configuration information for the SSH server.
show ssh	Shows the status of the SSH server.

For more information about these commands, see the "Secure Shell Commands" section in the "Other Security Features" chapter of the *Cisco IOS Security Command Reference, Cisco IOS Release 12.2,* at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_r/fothercr.htm.

RADIUS on the ML-Series Card

RADIUS is a distributed client/server system that secures networks against unauthorized access. Clients send authentication requests to a central RADIUS server, which contains all user authentication and network service access information. The RADIUS host is normally a multiuser system running RADIUS server software from Cisco or another software provider.

Many Cisco products offer RADIUS support, including the ONS 15454, ONS 15454 SDH, ONS 15310-CL, ONS 15310-MA, and ONS 15600. The ML-Series card also supports RADIUS.

The ML-Series card can operate either in RADIUS relay mode or in RADIUS stand alone mode (default). In either mode, the RADIUS messages from the ML-Series card are passed to a RADIUS server that is on the data communications network (DCN) used to manage the ONS node.

RADIUS Relay Mode

In RADIUS relay mode, RADIUS on the ML-Series card is configured by CTC or TL1 and uses the AAA/RADIUS features of the ONS node, which contains the ML-Series card. There is no interaction between RADIUS relay mode and RADIUS standalone mode. For information on ONS node security, refer to the "Security" chapter of the ONS node's reference manual.

An ML-Series card operating in RADIUS relay mode does need to be specified as a client in the RADIUS server entries. The RADIUS server uses the client entry for the ONS node as a proxy for the ML-Series card.

Enabling relay mode disables the Cisco IOS CLI commands used to configure AAA/RADIUS. The user can still use the Cisco IOS CLI commands not related to AAA/RADIUS.

In relay mode, the ML-Series card shows a RADIUS server host with an IP address that is really the internal IP address of the active timing, communications, and control card (XTC). When the ML-Series card actually sends RADIUS packets to this internal address, the XTC converts the RADIUS packet destination into the real IP address of the RADIUS server. In stand alone mode, the ML-Series card shows the true IP addresses of the RADIUS servers.

When in relay mode with multiple RADIUS server hosts, the ML-Series card IOS CLI **show run** output also shows the internal IP address of the active XTC card. But since the single IP address now represents multiple hosts, different port numbers are paired with the IP address to distinguish the individual hosts. These ports are from 1860 to 1869, one for each authentication server host configured, and from 1870 to 1879, one for each accounting server host configured.

The single IP address will not match the host IP addresses shown in CTC, which uses the true addresses of the RADIUS server hosts. These same true IP addresses appear in the ML-Series card IOS CLI **show run** output, when the ML-Series card is in stand alone mode.

Note A user can configure up to 10 servers for either authentication or accounting application, and one server host can perform both authentication and accounting applications.

Configuring RADIUS Relay Mode

This feature is turned on with CTC or TL1. To enable RADIUS Relay Mode through CTC, go to the card-level view of the ML-Series card, check the **Enable RADIUS Relay** box and click **Apply.** The user must be logged in at the Superuser level to complete this task.

To enable it using TL1, refer to the *Cisco ONS SONET TL1 Command Guide*.

Caution Switching the ML-Series card into RADIUS relay mode erases any configuration in the Cisco IOS configuration file related to AAA/RADIUS. The cleared AAA/RADIUS configuration is not restored to the Cisco IOS configuration file when the ML-Series card is put back into stand alone mode.

Caution Do not use the Cisco IOS command copy running-config startup-config while the ML-Series card is in relay mode. This command will save a Cisco IOS configuration file with RADIUS relay enabled. On a reboot, the ML-Series card would come up in RADIUS relay mode, even when the Enable RADIUS Relay box on the CTC is not checked. If this situation arises, the user should check the Enable RADIUS Relay box and click Apply and then uncheck the Enable RADIUS Relay box and click Apply. Doing this will set the ML-Series card in stand alone mode and clear RADIUS relay from the ML-Series card configuration.

RADIUS Stand Alone Mode

In stand alone mode, RADIUS on the ML-Series card is configured with the Cisco IOS CLI in the same general manner as RADIUS on a Cisco Catalyst switch.

This section describes how to enable and configure RADIUS in the stand alone mode on the ML-Series card. RADIUS in stand alone mode is facilitated through AAA and enabled through AAA commands.

Note For the remainder of the chapter, RADIUS refers to the Cisco IOS RADIUS available when the ML-Series card is in stand alone mode. It does not refer to RADIUS relay mode.

Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Security Command Reference, Release 12.2*.

These sections contain this configuration information:

Understanding RADIUS

When a user attempts to log in and authenticate to an ML-Series card with access controlled by a RADIUS server, these events occur:

1. The user is prompted to enter a username and password.
2. The username and encrypted password are sent over the network to the RADIUS server.
3. The user receives one of these responses from the RADIUS server:
   a. ACCEPT—The user is authenticated.
   b. REJECT—The user is either not authenticated and is prompted to reenter the username and password, or access is denied.

The ACCEPT and REJECT responses are bundled with additional data that is used for privileged EXEC or network authorization. Users must first successfully complete RADIUS authentication before proceeding to RADIUS authorization if it is enabled. The additional data included with the ACCEPT and REJECT packets includes these items:

- Telnet, SSH, rlogin, or privileged EXEC services
- Connection parameters, including the host or client IP address, access list, and user timeouts

Configuring RADIUS

This section describes how to configure your ML-Series card to support RADIUS. At a minimum, you must identify the host or hosts that run the RADIUS server software and define the method lists for RADIUS authentication. You must also apply the method list to the interface on which you want authentication to occur. For the ML-Series card, this is the vty ports. You can optionally define method lists for RADIUS authorization and accounting.

You should have access to and should configure a RADIUS server before configuring RADIUS features on your ML-Series card.

These sections contain this configuration information:

- Default RADIUS Configuration, page 15-9
- Identifying the RADIUS Server Host, page 15-9 (required)
- Configuring AAA Login Authentication, page 15-11 (required)
- Defining AAA Server Groups, page 15-13 (optional)
- Configuring RADIUS Authorization for User Privileged Access and Network Services, page 15-15 (optional)
- Starting RADIUS Accounting, page 15-16 (optional)
- Configuring a nas-ip-address in the RADIUS Packet, page 15-17 (optional)
- Configuring Settings for All RADIUS Servers, page 15-17 (optional)
- Configuring the ML-Series Card to Use Vendor-Specific RADIUS Attributes, page 15-18 (optional)
- Configuring the ML-Series Card for Vendor-Proprietary RADIUS Server Communication, page 15-19 (optional)

Default RADIUS Configuration

RADIUS and AAA are disabled by default. To prevent a lapse in security, you cannot configure RADIUS through a network management application. When enabled, RADIUS can authenticate users accessing the ML-Series card through the Cisco IOS CLI.

Identifying the RADIUS Server Host

ML-Series-card-to-RADIUS-server communication involves several components:

- Hostname or IP address
- Authentication destination port
- Accounting destination port
- Key string
- Timeout period
- Retransmission value

You identify RADIUS security servers by their hostname or IP address, their hostname and specific UDP port numbers, or their IP address and specific UDP port numbers. The combination of the IP address and the UDP port number creates a unique identifier, allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. This unique identifier enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address.

If two different host entries on the same RADIUS server are configured for the same service—for example, accounting—the second host entry configured acts as a fail-over backup to the first one. Using this example, if the first host entry fails to provide accounting services, the ML-Series card tries the second host entry configured on the same device for accounting services.

To configure RADIUS to use the AAA security commands, you must specify the host running the RADIUS server daemon and a secret text (key) string that it shares with the ML-Series card. A RADIUS server, the ONS node, and the ML-Series card use a shared secret text string to encrypt passwords and exchange responses. The system ensures that the ML-Series cards' shared secret matches the shared secret in the ONS node.

Note If you configure both global and per-server functions (timeout, retransmission, and key commands) on the switch, the per-server timer, retransmission, and key value commands override global timer, retransmission, and key value commands. For information on configuring these settings on all RADIUS servers, see the "Configuring Settings for All RADIUS Servers" section on page 15-17.

Note Retransmission and timeout period values can be configured on the ML-Series card in stand alone mode. These values cannot be configured on the ML-Series card in relay mode.

You can configure the ML-Series card to use AAA server groups to group existing server hosts for authentication. For more information, see the "Defining AAA Server Groups" section on page 15-13.

Beginning in privileged EXEC mode, follow these steps to configure per-server RADIUS server communication. This procedure is required.

	Command	Purpose
Step 1	**Router # configure terminal**	Enter global configuration mode.
Step 2	**Router (config)# aaa new-model**	Enable AAA.
Step 3	**Router (config)# radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]	Specify the IP address or hostname of the remote RADIUS server host. • (Optional) For **auth-port** *port-number*, specify the UDP destination port for authentication requests. • (Optional) For **acct-port** *port-number*, specify the UDP destination port for accounting requests. • (Optional) For **timeout** *seconds*, specify the time interval that the switch waits for the RADIUS server to reply before resending. The range is 1 to 1000. This setting overrides the **radius-server timeout** global configuration command setting. If no timeout is set with the **radius-server host** command, the setting of the **radius-server timeout** command is used. • (Optional) For **retransmit** *retries*, specify the number of times a RADIUS request is resent to a server if that server is not responding or responding slowly. The range is 1 to 1000. If no retransmit value is set with the **radius-server host** command, the setting of the **radius-server retransmit** global configuration command is used. • (Optional) For **key** *string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in the **radius-server host** command. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key. To configure the switch to recognize more than one host entry associated with a single IP address, enter this command as many times as necessary, making sure that each UDP port number is different. The switch software searches for hosts in the order in which you specify them. Set the timeout, retransmit, and encryption key values to use with the specific RADIUS host.
Step 4	**Router (config)# end**	Return to privileged EXEC mode.
Step 5	**Router# show running-config**	Verify your entries.
Step 6	**Router# copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the specified RADIUS server, use the **no radius-server host** *hostname* | *ip-address* global configuration command.

This example shows how to configure one RADIUS server to be used for authentication and another to be used for accounting:

```
Switch(config)# radius-server host 172.29.36.49 auth-port 1612 key rad1
Switch(config)# radius-server host 172.20.36.50 acct-port 1618 key rad2
```

This example shows how to configure *host1* as the RADIUS server and to use the default ports for both authentication and accounting:

```
Switch(config)# radius-server host host1
```

Note You also need to configure some settings on the RADIUS server. These settings include the IP address of the switch and the key string to be shared by both the server and the switch. For more information, see the RADIUS server documentation.

Configuring AAA Login Authentication

To configure AAA authentication, you define a named list of authentication methods and then apply that list to various ports. The method list defines the types of authentication to be performed and the sequence in which they are performed; it must be applied to a specific port before any of the defined authentication methods are performed. The only exception is the default method list, which is named *default*. The default method list is automatically applied to all ports except those that have a named method list explicitly defined.

A method list describes the sequence and authentication methods to be queried to authenticate a user. You can designate one or more security protocols to be used for authentication, thus ensuring a backup system for authentication in case the initial method fails. The software uses the first method listed to authenticate users; if that method fails to respond, the software selects the next authentication method in the method list. This process continues until there is successful communication with a listed authentication method or until all defined methods are exhausted. If authentication fails at any point in this cycle—meaning that the security server or local username database responds by denying the user access—the authentication process stops, and no other authentication methods are attempted.

For additional information on AAA login, refer to the "Authentication, Authorization, and Accounting (AAA)" chapter of the *Cisco IOS Security Configuration Guide, Release 12.2* at: http://www.cisco.com/en/US/products/sw/iosswrel/ps1835/products_installation_and_configuration_guides_list.html

Beginning in privileged EXEC mode, follow these steps to configure login authentication. This procedure is required.

	Command	Purpose
Step 1	**Router# configure terminal**	Enter global configuration mode.
Step 2	**Router (config)# aaa new-model**	Enable AAA.

	Command	Purpose
Step 3	**Router (config)# aaa authentication login** {**default** | *list-name*} *method1* [*method2...*]	Create a login authentication method list. • To create a default list that is used when a named list is *not* specified in the **login authentication** command, use the **default** keyword followed by the methods that are to be used in default situations. The default method list is automatically applied to all ports. • For *list-name*, specify a character string to name the list you are creating. • For *method1...*, specify the actual method the authentication algorithm tries. The additional methods of authentication are used only if the previous method returns an error, not if it fails. Select one of these methods: – **enable**—Use the enable password for authentication. Before you can use this authentication method, you must define an enable password by using the **enable** *password* global configuration command. – **group radius**—Use RADIUS authentication. Before you can use this authentication method, you must configure the RADIUS server. For more information, see the "Identifying the RADIUS Server Host" section on page 15-9. – **line**—Use the line password for authentication. Before you can use this authentication method, you must define a line password. Use the **password** *password* line configuration command. – **local**—Use the local username database for authentication. You must enter username information in the database. Use the **username** *name* **password** global configuration command. – **local-case**—Use a case-sensitive local username database for authentication. You must enter username information in the database by using the **username** *password* global configuration command. – **none**—Do not use any authentication for login.
Step 4	**Router (config)# line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*]	Enter line configuration mode, and configure the lines to which you want to apply the authentication list.

	Command	Purpose
Step 5	**Router (config-line)# login authentication** {**default** | *list-name*}	Apply the authentication list to a line or set of lines. • If you specify **default**, use the default list created with the **aaa authentication login** command. • For *list-name*, specify the list created with the **aaa authentication login** command.
Step 6	**Router (config)# end**	Return to privileged EXEC mode.
Step 7	**Router# show running-config**	Verify your entries.
Step 8	**Router# copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable AAA, use the **no aaa new-model** global configuration command. To disable AAA authentication, use the **no aaa authentication login** {**default** | *list-name*} *method1* [*method2*...] global configuration command. To either disable RADIUS authentication for logins or to return to the default value, use the **no login authentication** {**default** | *list-name*} line configuration command.

Defining AAA Server Groups

You can configure the ML-Series card to use AAA server groups to group existing server hosts for authentication. You select a subset of the configured server hosts and use them for a particular service. The server group is used with a global server-host list, which lists the IP addresses of the selected server hosts.

Server groups also can include multiple host entries for the same server if each entry has a unique identifier (the combination of the IP address and UDP port number), allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. If you configure two different host entries on the same RADIUS server for the same service (for example, accounting), the second configured host entry acts as a fail-over backup to the first one.

You use the **server** group server configuration command to associate a particular server with a defined group server. You can either identify the server by its IP address or identify multiple host instances or entries by using the optional **auth-port** and **acct-port** keywords.

Beginning in privileged EXEC mode, follow these steps to define the AAA server group and associate a particular RADIUS server with it:

	Command	Purpose
Step 1	**Router# configure terminal**	Enter global configuration mode.
Step 2	**Router (config)# aaa new-model**	Enable AAA.

	Command	Purpose
Step 3	**Router (config)# radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]	Specify the IP address or hostname of the remote RADIUS server host. • (Optional) For **auth-port** *port-number*, specify the UDP destination port for authentication requests. • (Optional) For **acct-port** *port-number*, specify the UDP destination port for accounting requests. • (Optional) For **timeout** *seconds*, specify the time interval that the switch waits for the RADIUS server to reply before resending. The range is 1 to 1000. This setting overrides the **radius-server timeout** global configuration command setting. If no timeout is set with the **radius-server host** command, the setting of the **radius-server timeout** command is used. • (Optional) For **retransmit** *retries*, specify the number of times a RADIUS request is resent to a server if that server is not responding or responding slowly. The range is 1 to 1000. If no retransmit value is set with the **radius-server host** command, the setting of the **radius-server retransmit** global configuration command is used. • (Optional) For **key** *string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in the **radius-server host** command. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key. To configure the switch to recognize more than one host entry associated with a single IP address, enter this command as many times as necessary, making sure that each UDP port number is different. The switch software searches for hosts in the order in which you specify them. Set the timeout, retransmit, and encryption key values to use with the specific RADIUS host.
Step 4	**Router (config)# aaa group server radius** *group-name*	Define the AAA server-group with a group name. This command puts the ML-Series card in a server group configuration mode.
Step 5	**Router (config-sg-radius)# server** *ip-address*	Associate a particular RADIUS server with the defined server group. Repeat this step for each RADIUS server in the AAA server group. Each server in the group must be previously defined in Step 2.
Step 6	**Router (config-sg-radius)# end**	Return to privileged EXEC mode.
Step 7	**Router # show running-config**	Verify your entries.
Step 8	**Router # copy running-config startup-config**	(Optional) Save your entries in the configuration file.
Step 9		Enable RADIUS login authentication. See the "Configuring AAA Login Authentication" section on page 15-11.

To remove the specified RADIUS server, use the **no radius-server host** *hostname* | *ip-address* global configuration command. To remove a server group from the configuration list, use the **no aaa group server radius** *group-name* global configuration command. To remove the IP address of a RADIUS server, use the **no server** *ip-address* server group configuration command.

In this example, the ML-Series card is configured to recognize two different RADIUS group servers (*group1* and *group2*). Group1 has two different host entries on the same RADIUS server configured for the same services. The second host entry acts as a fail-over backup to the first entry.

```
Switch(config)# radius-server host 172.20.0.1 auth-port 1000 acct-port 1001
Switch(config)# radius-server host 172.10.0.1 auth-port 1645 acct-port 1646
Switch(config)# aaa new-model
Switch(config)# aaa group server radius group1
Switch(config-sg-radius)# server 172.20.0.1 auth-port 1000 acct-port 1001
Switch(config-sg-radius)# exit
Switch(config)# aaa group server radius group2
Switch(config-sg-radius)# server 172.20.0.1 auth-port 2000 acct-port 2001
Switch(config-sg-radius)# exit
```

Configuring RADIUS Authorization for User Privileged Access and Network Services

AAA authorization limits the services available to a user. When AAA authorization is enabled, the ML-Series card uses information retrieved from the user's profile, which is in the local user database or on the security server, to configure the user's session. The user is granted access to a requested service only if the information in the user profile allows it.

There is no support for setting the privilege level on the ML-Series card or using the **priv-lvl** command. A user authenticating with a RADIUS server will only access the ML-Series card with a privilege level of 1, which is the default login privilege level. Because of this, a priv-lvl configured on the RADIUS server should have the priv-lvl of 0 or 1. Once a user is authenticated and gains access to the ML-Series card, they can use the enable password to gain privileged EXEC authorization and become a super user with a privilege level of 15, which is the default privilege level of enable mode.

This example of an ML-Series card user record is from the output of the RADIUS server and shows the privilege level:

```
CISCO15 Auth-Type := Local, User-Password == "otbu+1"
Service-Type = Login,
Session-Timeout = 100000,
Cisco-AVPair = "shell:priv-lvl=1"
```

You can use the **aaa authorization** global configuration command with the **radius** keyword to set parameters that restrict a user's network access to privileged EXEC mode.

The **aaa authorization exec radius local** command sets these authorization parameters:

- Use RADIUS for privileged EXEC access authorization if authentication was performed by using RADIUS.
- Use the local database if authentication was not performed by using RADIUS.

Note Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

Beginning in privileged EXEC mode, follow these steps to specify RADIUS authorization for privileged EXEC access and network services:

	Command	Purpose
Step 1	**Router# configure terminal**	Enter global configuration mode.
Step 2	**Router (config)# aaa authorization network radius**	Configure the ML-Series card for user RADIUS authorization for all network-related service requests.
Step 3	**Router (config)# aaa authorization exec radius**	Configure the ML-Series card for user RADIUS authorization if the user has privileged EXEC access. The **exec** keyword might return user profile information (such as **autocommand** information).
Step 4	**Router (config)# end**	Return to privileged EXEC mode.
Step 5	**Router# show running-config**	Verify your entries.
Step 6	**Router# copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable authorization, use the **no aaa authorization** {**network** | **exec**} *method1* global configuration command.

Starting RADIUS Accounting

The AAA accounting feature tracks the services that users are accessing and the amount of network resources that they are consuming. When AAA accounting is enabled, the ML-Series card reports user activity to the RADIUS security server in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server. This data can then be analyzed for network management, client billing, or auditing.

Beginning in privileged EXEC mode, follow these steps to enable RADIUS accounting for each Cisco IOS privilege level and for network services:

	Command	Purpose
Step 1	**Router# configure terminal**	Enter global configuration mode.
Step 2	**Router (config)# aaa accounting network start-stop radius**	Enable RADIUS accounting for all network-related service requests.
Step 3	**Router (config)# aaa accounting exec start-stop radius**	Enable RADIUS accounting to send a start-record accounting notice at the beginning of a privileged EXEC process and a stop-record at the end.
Step 4	**Router (config)# end**	Return to privileged EXEC mode.
Step 5	**Router# show running-config**	Verify your entries.
Step 6	**Router# copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable accounting, use the **no aaa accounting** {**network** | **exec**} **start-stop** *method1...* global configuration command.

Configuring a nas-ip-address in the RADIUS Packet

The ML-Series card in RADIUS relay mode allows the user to configure a separate nas-ip-address for each ML-Series card. In RADIUS standalone mode, this command is hidden in the Cisco IOS CLI. This allows the RADIUS server to distinguish among individual ML-Series card in the same ONS node. Identifying the specific ML-Series card that sent the request to the server can be useful in debugging from the server. The nas-ip-address is primarily used for validation of the RADIUS authorization and accounting requests.

If this value is not configured, the nas-ip-address is filled in by the normal Cisco IOS mechanism using the value configured by the **ip radius-source** command. If no value is specified then the best IP address routable to the server is used. If no routable address is available, the IP address of the server is used.

Beginning in privileged EXEC mode, follow these steps to configure the nas-ip-address:

	Command	Purpose
Step 1	**Router# configure terminal**	Enter global configuration mode.
Step 2	**Router (config)#** [no] ip radius nas-ip-address {*hostname* | *ip-address*}	Specify the IP address or hostname of the attribute 4 (nas-ip-address) in the radius packet. If there is only one ML-Series card in the ONS node, this command does not provide any advantage. The public IP address of the ONS node serves as the nas-ip-address in the RADIUS packet sent to the server.
Step 3	**Router (config)# end**	Return to privileged EXEC mode.
Step 4	**Router# show running-config**	Verify your settings.
Step 5	**Router# copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring Settings for All RADIUS Servers

Beginning in privileged EXEC mode, follow these steps to configure global communication settings between the ML-Series card and all RADIUS servers:

	Command	Purpose
Step 1	**Router# configure terminal**	Enter global configuration mode.
Step 2	**Router (config)# radius-server key** *string*	Specify the shared secret text string used between the ML-Series card and all RADIUS servers. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.
Step 3	**Router (config)# radius-server retransmit** *retries*	Specify the number of times the ML-Series card sends each RADIUS request to the server before giving up. The default is 3; the range 1 to 1000.
Step 4	**Router (config)# radius-server timeout** *seconds*	Specify the number of seconds a ML-Series card waits for a reply to a RADIUS request before resending the request. The default is 5 seconds; the range is 1 to 1000.

	Command	Purpose
Step 5	**Router (config)# radius-server deadtime** *minutes*	Specify the number of minutes to mark as "dead" any RADIUS servers that fail to respond to authentication requests. A RADIUS server marked as "dead" is skipped by additional authentication requests for the specified number of *minutes*. This allows trying the next configured server without having to wait for the request to time out before. If all RADIUS servers are marked as "dead," the skipping will not take place. The default is 0; the range is 1 to 1440 minutes.
Step 6	**Router (config)# end**	Return to privileged EXEC mode.
Step 7	**Router# show running-config**	Verify your settings.
Step 8	**Router# copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting for the retransmit, timeout, and deadtime, use the **no** forms of these commands.

Configuring the ML-Series Card to Use Vendor-Specific RADIUS Attributes

The Internet Engineering Task Force (IETF) draft standard specifies a method for communicating vendor-specific information between the ML-Series card and the RADIUS server by using the vendor-specific attribute (attribute 26). Vendor-specific attributes (VSAs) allow vendors to support their own extended attributes that are not suitable for general use. The Cisco RADIUS implementation supports one vendor-specific option by using the format recommended in the specification. Cisco's vendor-ID is 9, and the supported option has vendor-type 1, which is named *cisco-avpair*. The value is a string with this format:

```
protocol : attribute sep value *
```

Protocol is a value of the Cisco protocol attribute for a particular type of authorization. *Attribute* and *value* are an appropriate attribute-value (AV) pair defined in the Cisco Terminal Access Controller Access Control System Plus (TACACS+) specification, and *sep* is the character = for mandatory attributes and the character * for optional attributes. The full set of features available for TACACS+ authorization can then be used for RADIUS.

For example, this AV pair activates Cisco's *multiple named ip address pools* feature during IP authorization (during point-to-point protocol [PPP] internet protocol control protocol (IPCP) address assignment):

```
cisco-avpair= "ip:addr-pool=first"
```

This example shows how to specify an authorized VLAN in the RADIUS server database:

```
cisco-avpair= "tunnel-type(#64)=VLAN(13)"
cisco-avpair= "tunnel-medium-type(#65)=802 media(6)"
cisco-avpair= "tunnel-private-group-ID(#81)=vlanid"
```

This example shows how to apply an input access control list (ACL) in ASCII format to an interface for the duration of this connection:

```
cisco-avpair= "ip:inacl#1=deny ip 10.10.10.10 0.0.255.255 20.20.20.20 255.255.0.0"
cisco-avpair= "ip:inacl#2=deny ip 10.10.10.10 0.0.255.255 any"
cisco-avpair= "mac:inacl#3=deny any any decnet-iv"
```

This example shows how to apply an output ACL in ASCII format to an interface for the duration of this connection:

```
cisco-avpair= "ip:outacl#2=deny ip 10.10.10.10 0.0.255.255 any"
```

Other vendors have their own unique vendor-IDs, options, and associated VSAs. For more information about vendor-IDs and VSAs, see RFC 2138, "Remote Authentication Dial-In User Service (RADIUS)."

Beginning in privileged EXEC mode, follow these steps to configure the ML-Series card to recognize and use VSAs:

	Command	Purpose
Step 1	Router# **configure terminal**	Enter global configuration mode.
Step 2	Router (config)# **radius-server vsa send** [**accounting** | **authentication**]	Enable the ML-Series card to recognize and use VSAs as defined by RADIUS IETF attribute 26. • (Optional) Use the **accounting** keyword to limit the set of recognized vendor-specific attributes to only accounting attributes. • (Optional) Use the **authentication** keyword to limit the set of recognized vendor-specific attributes to only authentication attributes. If you enter this command without keywords, both accounting and authentication vendor-specific attributes are used. The AAA server includes the authorization level in the VSA response message for the ML-Series card.
Step 3	**Router (config)# end**	Return to privileged EXEC mode.
Step 4	**Router# show running-config**	Verify your settings.
Step 5	**Router# copy running-config startup-config**	(Optional) Save your entries in the configuration file.

For a complete list of RADIUS attributes or more information about vendor-specific attribute 26, see the "RADIUS Attributes" appendix in the *Cisco IOS Security Configuration Guide, Release 12.2*.

Configuring the ML-Series Card for Vendor-Proprietary RADIUS Server Communication

Although an IETF draft standard for RADIUS specifies a method for communicating vendor-proprietary information between the ML-Series card and the RADIUS server, some vendors have extended the RADIUS attribute set in a unique way. Cisco IOS software supports a subset of vendor-proprietary RADIUS attributes.

As mentioned earlier, to configure RADIUS (whether vendor-proprietary or IETF draft-compliant), you must specify the host running the RADIUS server daemon and the secret text string it shares with the ML-Series card. You specify the RADIUS host and secret text string by using the **radius-server** global configuration commands.

Beginning in privileged EXEC mode, follow these steps to specify a vendor-proprietary RADIUS server host and a shared secret text string:

	Command	Purpose
Step 1	**Router# configure terminal**	Enter global configuration mode.
Step 2	**Router (config)# radius-server host** {*hostname* | *ip-address*} **non-standard**	Specify the IP address or hostname of the remote RADIUS server host and identify that it is using a vendor-proprietary implementation of RADIUS.
Step 3	**Router (config)# radius-server key** *string*	Specify the shared secret text string used between the ML-Series card and the vendor-proprietary RADIUS server. The ML-Series card and the RADIUS server use this text string to encrypt passwords and exchange responses. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.
Step 4	**Router (config)# end**	Return to privileged EXEC mode.
Step 5	**Router# show running-config**	Verify your settings.
Step 6	**Router# copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete the vendor-proprietary RADIUS host, use the **no radius-server host** {*hostname* | *ip-address*} **non-standard** global configuration command. To disable the key, use the **no radius-server key** global configuration command.

This example shows how to specify a vendor-proprietary RADIUS host and to use a secret key of *rad124* between the ML-Series card and the server:

```
Switch(config)# radius-server host 172.20.30.15 nonstandard
Switch(config)# radius-server key rad124
```

Displaying the RADIUS Configuration

To display the RADIUS configuration, use the **show running-config** privileged EXEC command.

CHAPTER 16

Configuring Bridging on the ML-Series Card

This chapter describes how to configure bridging for the ML-Series card. Bridging is one of the simplest configurations of the ML-Series card. Other alternatives exist to simple bridging, such as Integrated Routing and Bridging (IRB). The user should consult the chapter detailing their desired type of configuration.

This chapter includes the following major sections:

Caution Cisco Inter-Switch Link (ISL) and Cisco Dynamic Trunking Protocol (DTP) are not supported by the ML-Series cards, but the ML-Series broadcast forwards these formats. Using ISL or DTP on connecting devices is not recommended. Some Cisco devices attempt to use ISL or DTP by default.

Understanding Bridging

The ML-Series card supports transparent bridging for Fast Ethernet, Fast EtherChannel (FEC), packet-over-SONET/SDH (POS) ports, and POS channel. It supports a maximum of 255 active bridge groups. Transparent bridging combines the speed and protocol transparency of a spanning-tree bridge, along with the functionality, reliability, and security of a router.

To configure bridging, you must perform the following tasks in the modes indicated:

- In global configuration mode:
  - Enable bridging of IP packets.
  - (Optional) Select the type of Spanning Tree Protocol (STP).
- In interface configuration mode:
  - Determine which interfaces belong to the same bridge group.

    The ML-Series card bridges all nonrouted traffic among the network interfaces comprising the bridge group. If spanning tree is enabled, the interfaces become part of the same spanning tree. Interfaces that do not participate in a bridge group cannot forward bridged traffic.

If the destination address of the packet is known in the bridge table, the packet is forwarded on a single interface in the bridge group. If the packet's destination is unknown in the bridge table, the packet is flooded on all forwarding interfaces in the bridge group. The bridge places source addresses in the bridge table as it learns them during the process of bridging.

Spanning tree is not mandatory for an ML-Series card bridge group, but if it is configured, a separate spanning-tree process runs for each configured bridge group. A bridge group establishes a spanning tree based on the bridge protocol data units (BPDUs) it receives on only its member interfaces.

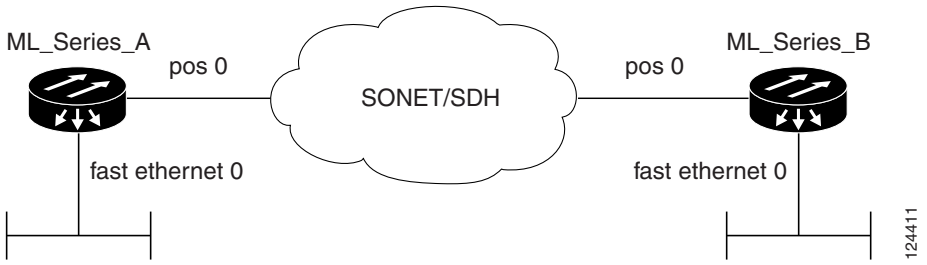
Configuring Bridging

Beginning in global configuration mode, use the following steps to configure bridging:

	Command	Purpose
Step 1	`ML_Series(config)# no ip routing`	Enables bridging of IP packets. This command needs to be executed once per card, not once per bridge-group. This step is not done for IRB.
Step 2	`ML_Series(config)# bridge` *bridge-group-number* `[protocol {drpi-rstp | rstp | ieee}]`	Assigns a bridge group number and defines the appropriate spanning-tree type: • **drpri-rstp** is the protocol used to interconnect dual resilient packet ring (RPR) to protect from node failure. Do not configure this option on the ONS 15310-CL or ONS 15310-MA ML-Series. • **rstp** is the IEEE 802.1W Rapid Spanning Tree. • **ieee** is the IEEE 802.1D Spanning Tree Protocol. **Note** Spanning tree is not mandatory for an ML-Series card bridge group, but configuring spanning tree blocks network loops.
Step 3	`ML_Series(config)# bridge` *bridge-group-number* `priority` *number*	(Optional) Assigns a specific priority to the bridge, to assist in the spanning-tree root definition. Lowering the priority of a bridge makes it more likely that the bridge is selected as the root.
Step 4	`ML_Series(config)# interface` *type number*	Enters interface configuration mode to configure the interface of the ML-Series card.
Step 5	`ML_Series(config-if)# bridge-group` *bridge-group-number*	Assigns a network interface to a bridge group.
Step 6	`ML_Series(config-if)# no shutdown`	Changes the shutdown state to up and enables the interface.
Step 7	`ML_Series(config-if)# end`	Returns to privileged EXEC mode.
Step 8	`ML_Series# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Figure 16-1 shows a bridging example. Example 16-1 shows the code used to configure ML-Series A. Example 16-2 shows the code used to configure ML-Series B.

Figure 16-1 Bridging Example



Example 16-1 ML_Series A Configuration

```
bridge irb
bridge 1 protocol ieee
!
!
interface FastEthernet0
 no ip address
 bridge-group 1
!
interface POS0
 no ip address
bridge-group 1
```

Example 16-2 ML_Series B Configuration

```
bridge irb
bridge 1 protocol ieee
!
!
interface FastEthernet0
 no ip address
 bridge-group 1
!
interface POS0
 no ip address
bridge-group 1
```

Monitoring and Verifying Bridging

After you have set up the ML-Series card for bridging, you can monitor and verify its operation by performing the following procedure in privileged EXEC mode:

	Command	Purpose
Step 1	ML_Series# **clear bridge** *bridge-group-number*	Removes any learned entries from the forwarding database of a particular bridge group, clears the transmit, and receives counts for any statically configured forwarding entries.
Step 2	ML_Series# **show bridge** {*bridge-group-number* | *interface-address*}	Displays classes of entries in the bridge forwarding database.

	Command	Purpose
Step 3	`ML_Series# show bridge verbose`	Displays detailed information about configured bridge groups.
Step 4	`ML_Series# show spanning-tree [bridge-group-number][brief]`	Displays detailed information about spanning tree. • ***bridge-group-number*** restricts the spanning tree information to specific bridge groups. • **brief** displays summary information about spanning tree.

Example 16-3 shows examples of monitoring and verifying bridging.

Example 16-3 Monitoring and Verifying Bridging

```
ML_Series# show bridge 1

Total of 1260 station blocks, 310 free
Codes: P - permanent, S - self

Bridge Group 1:

Maximum dynamic entries allowed: 1000
Current dynamic entry count: 1

    Address       Action   Interface
0000.0001.3100   forward   FastEthernet0

ML_Series# show spanning-tree 1
Bridge group 1 is executing the rstp compatible Spanning Tree protocol
  Bridge Identifier has priority 32768, sysid 1, address 000b.fcfa.339e
  Configured hello time 2, max age 20, forward delay 15
  We are the root of the spanning tree
  Topology change flag not set, detected flag not set
Number of topology changes 1 last change occurred 1w1d ago
          from POS0.1
  Times:  hold 1, topology change 35, notification 2
          hello 2, max age 20, forward delay 15
  Timers: hello 0, topology change 0, notification 0, aging 300

 Port 3 (FastEthernet0) of Bridge group 1 is designated disabled
   Port path cost 19, Port priority 128, Port Identifier 128.3.
   Designated root has priority 32769, address 000b.fcfa.339e
   Designated bridge has priority 32769, address 000b.fcfa.339e
   Designated port id is 128.3, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 0
   Link type is point-to-point by default
   BPDU: sent 0, received 0

 Port 15 (POS0.1) of Bridge group 1 is designated down
   Port path cost 37, Port priority 128, Port Identifier 128.15.
   Designated root has priority 32769, address 000b.fcfa.339e
   Designated bridge has priority 32769, address 000b.fcfa.339e
   Designated port id is 128.15, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default
   BPDU: sent 370832, received 4
```

CHAPTER 17

CE-100T-8 Ethernet Operation

This chapter describes the operation of the CE-100T-8 (Carrier Ethernet) card supported on the Cisco ONS 15310-CL and ONS 15310-MA (15310-CE-100T-8). A CE-100T-8 card is also supported on the ONS 15454 (15454-CE-100T-8). Provisioning is done through Cisco Transport Controller (CTC) or Transaction Language One (TL1). Cisco IOS is not supported on the CE-100T-8 card.

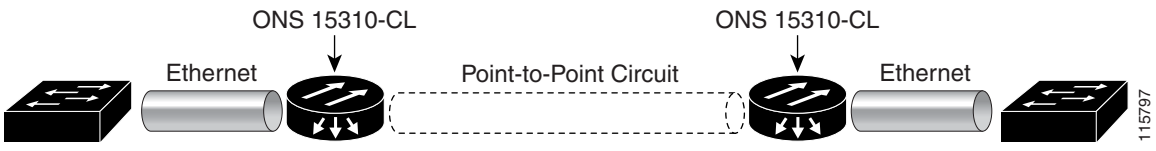
For Ethernet card specifications, refer to the *Cisco ONS 15454 Reference Manual*. For step-by-step Ethernet card circuit configuration procedures and hard-reset and soft-reset procedures, refer to the *Cisco ONS 15454 Procedure Guide*. For TL1 provisioning commands, refer to the *Cisco ONS SONET TL1 Command Guide*. For specific details on ONS 15310-CL and ONS 15310-MA Ethernet card interoperability with other ONS platforms, refer to the "POS on ONS Ethernet Cards" chapter of the *Cisco ONS 15454 and Cisco ONS 15454 SDH Ethernet Card Software Feature and Configuration Guide.*

Chapter topics include:

CE-100T-8 Overview

The CE-100T-8 is a Layer 1 mapper card with eight 10/100 Ethernet ports. It maps each port to a unique SONET circuit in a point-to-point configuration. Figure 17-1 illustrates a sample CE-100T-8 application. In this example, data traffic from the Fast Ethernet port of a switch travels across the point-to-point circuit to the Fast Ethernet port of another switch.

Figure 17-1 CE-100T-8 Point-to-Point Circuit



The CE-100T-8 cards allow you to provision and manage an Ethernet private line service like a traditional SONET line. CE-100T-8 card applications include providing Ethernet private line services and high-availability transport. It supports ITU-T G.707 and Telcordia GR-253 based standards for SONET.

The CE-100T-8 offers full TL1-based provisioning capability. Refer to the *Cisco ONS SONET TL1 Command Guide* for CE-100T-8 TL1 provisioning commands.

CE-100T-8 Ethernet Features

The CE-100T-8 card has eight front-end Ethernet ports which use standard RJ-45 connectors for 10BASE-T Ethernet/100BASE-TX Ethernet media. Ethernet Ports 1 through 8 each map to a POS port with a corresponding number. The console port on the CE-100T-8 card is not functional.

The CE-100T-8 cards forward valid Ethernet frames unmodified over the SONET network. Information in the headers is not affected by the encapsulation and transport. For example, included IEEE 802.1Q information will travel through the process unaffected.

The ONS 15454 CE-100T-8 and the ONS 15310 CE-100T-8 support maximum Ethernet frame sizes of 1600 bytes including the CRC. The MTU size is not configureable and is set at a 1500 byte maximum (standard Ethernet MTU). Baby giant frames in which the standard Ethernet frame is augmented by IEEE 802.1 Q tags or MPLS tags are also supported. Full Jumbo frames (9000 byte maximum) are not supported.

The CE-100T-8 cards discard certain types of erroneous Ethernet frames rather than transport them over SONET. Erroneous Ethernet frames include corrupted frames with cyclic redundancy check (CRC) errors and undersized frames that do not conform to the minimum 64-byte length Ethernet standard.

Note Many Ethernet attributes are also available through the network element default feature. For more information on NE defaults, refer to the "Network Element Defaults" appendix in the *Cisco ONS 15454 Reference Manual*.

Autonegotiation, Flow Control, and Frame Buffering

On the CE-100T-8 card, Ethernet link autonegotiation is on by default when the speed or duplex of the port is set to auto. The user can also set the link speed, duplex, selective autonegotiation, and flow control manually under the card-level Provisioning tab of CTC.

The CE-100T-8 card supports selective autonegotiation on the Ethernet ports. If selective autonegotiation is enabled, the port attempts to autonegotiate only to a specific speed and duplex. The link will come up if both the speed and duplex of the attached autonegotiating device matches that of the port. You cannot enable selective autonegotiation if either the speed or duplex of the port is set to auto.

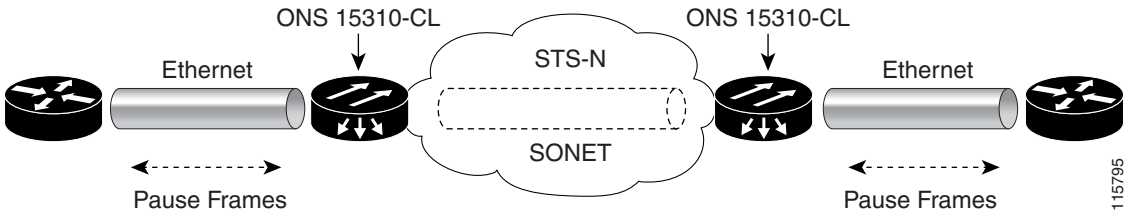
The CE-100T-8 card supports IEEE 802.3x flow control and frame buffering to reduce data traffic congestion. Flow control is on by default.

To prevent over-subscription, buffer memory is available for each port. When the buffer memory on the Ethernet port nears capacity, the CE-100T-8 card uses IEEE 802.3x flow control to transmit a pause frame to the attached Ethernet device. Flow control and autonegotiation frames are local to the Fast Ethernet interfaces and the attached Ethernet devices. These frames do not continue through the POS ports.

The CE-100T-8 card has symmetric flow control and proposes symmetric flow control when autonegotiating flow control with attached Ethernet devices. Symmetric flow control allows the CE-100T-8 cards to respond to pause frames sent from external devices and to send pause frames to external devices.

The pause frame instructs the source to stop sending packets for a specific period of time. The sending station waits the requested amount of time before sending more data. Figure 17-2 illustrates pause frames being sent and received by CE-100T-8 cards and attached switches.

Figure 17-2 Flow Control



This flow-control mechanism matches the sending and receiving device throughput to that of the bandwidth of the STS circuit. For example, a router might transmit to the Ethernet port on the CE-100T-8 card. This particular data rate might occasionally exceed 51.84 Mbps, but the SONET circuit assigned to the CE-100T-8 port might be only STS-1 (51.84 Mbps). In this example, the CE-100T-8 sends out a pause frame and requests that the router delay its transmission for a certain period of time. With flow control and a substantial per-port buffering capability, a private line service provisioned at less than full line rate capacity (STS-1) is efficient because frame loss can be controlled to a large extent.

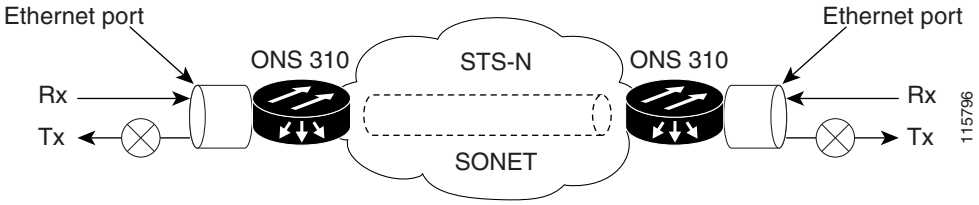
Ethernet Link Integrity Support

The CE-100T-8 supports end-to-end Ethernet link integrity (Figure 17-3). This capability is integral to providing an Ethernet private line service and correct operation of Layer 2 and Layer 3 protocols on the attached Ethernet devices.

End-to-end Ethernet link integrity means that if any part of the end-to-end path fails, the entire path fails. It disables the Ethernet port on the CE-100T-8 card if the remote Ethernet port is unable to transmit over the SONET network or if the remote Ethernet port is disabled.

Failure of the entire path is ensured by turning off the transmit pair at each end of the path. The attached Ethernet devices recognize the disabled transmit pair as a loss of carrier and consequently an inactive link or link fail.

Figure 17-3 End-to-End Ethernet Link Integrity Support



Note Some network devices can be configured to ignore a loss of carrier condition. If a device configured to ignore a loss of carrier condition attaches to a CE-100T-8 card at one end, alternative techniques (such as use of Layer 2 or Layer 3 keep-alive messages) are required to route traffic around failures. The response time of such alternate techniques is typically much longer than techniques that use link state as indications of an error condition.

Enhanced State Model for Ethernet and SONET Ports

The CE-100T-8 supports the Enhanced State Model (ESM) for the Ethernet ports, as well as for the SONET circuit. For more information about the ESM, refer to the "Administrative and Service States" appendix in the *Cisco ONS 15454 Reference Manual*.

The Ethernet ports can be set to the ESM service states including the In-Service, Automatic In-Service (IS,AINS) administrative state. IS,AINS initially puts the port in the Out-of-Service and Autonomous, Automatic In-Service (OOS-AU,AINS) state. In this service state, alarm reporting is suppressed, but traffic is carried and loopbacks are allowed. After the soak period passes, the port changes to In-Service and Normal (IS-NR). Raised fault conditions, whether their alarms are reported or not, can be retrieved on the CTC Conditions tab or by using the TL1 RTRV-COND command.

Two Ethernet port alarms/conditions, CARLOSS and TPTFAIL, can prevent the port from going into service. This occurs even though alarms are suppressed when a CE-100T-8 circuit is provisioned with the Ethernet ports set to the IS,AINS state, because the CE-100T-8 link integrity function is active and ensures that the links at both ends are not enabled until all SONET and Ethernet errors along the path are cleared. As long as the link integrity function keeps the end-to-end path down, both ports will have at least one of the two conditions needed to suppress the AINS-to-IS transition. Therefore, the ports will remain in the AINS state with alarms suppressed.

ESM also applies to the SONET circuits of the CE-100T-8 card. If the SONET circuit is set up in IS,AINS state and the Ethernet error occurs before the circuit transitions to IS, then link integrity will also prevent the circuit transition to the IS state until the Ethernet port errors are cleared at both ends. The service state will be OOS-AU,AINS as long as the administrative state is IS,AINS. When there are no Ethernet or SONET errors, link integrity enables the Ethernet port at each end. Simultaneously, the AINS countdown begins as normal. If no additional conditions occur during the time period, each port transitions to the IS-NR state. During the AINS countdown, the soak time remaining is available in CTC and TL1. The AINS soaking logic restarts from the beginning if a condition appears again during the soak period.

A SONET circuit provisioned in the IS,AINS state remains in the initial Out-of-Service (OOS) state until the Ethernet ports on each end of the circuit transition to the IS-NR state. The SONET circuit transports Ethernet traffic and counts statistics when link integrity turns on the Ethernet port, regardless of whether this AINS-to-IS transition is complete.

IEEE 802.1Q CoS and IP ToS Queuing

The CE-100T-8 references IEEE 802.1Q class of service (CoS) thresholds and IP type of service (ToS) (IP Differentiated Services Code Point [DSCP]) thresholds for priority queueing. CoS and ToS thresholds for the CE-100T-8 are provisioned on a per port level. This allows the user to provide priority treatment based on open standard quality of service (QoS) schemes already existing in the data network attached to the CE-100T-8. The QoS treatment is applied to both Ethernet and POS ports.

Any packet or frame with a priority greater than the set threshold is treated as priority traffic. This priority traffic is sent to the priority queue instead of the normal queue. When buffering occurs, packets on the priority queue preempt packets on the normal queue. This results in lower latency for the priority traffic, which is often latency-sensitive traffic, such as VoIP.

Because these priorities are placed on separate queues, the priority queuing feature should not be used to separate rate-based CIR/EIR marked traffic (sometimes done at a Metro Ethernet service provider edge). This could result in out-of-order packet delivery for packets of the same application, which would cause performance issues with some applications.

For an IP ToS-tagged packet, the CE-100T-8 can map any of the 256 priorities specified in IP ToS to priority or best effort. The user can configure a different ToS on CTC at the card-level view under the Provisioning > Ether Ports tabs. Any ToS class higher than the class specified in CTC is mapped to the priority queue, which is the queue geared towards low latency. By default, the ToS is set to 255, which is the highest ToS value. This results in all traffic being treated with equal priority by default.

Table 17-3 shows which values are mapped to the priority queue for sample IP ToS settings. (ToS settings span the full 0 to 255 range, but only selected settings are shown.)

Table 17-1 IP ToS Priority Queue Mappings

ToS Setting in CTC	ToS Values Sent to Priority Queue
255 (default)	None
250	251–255
150	151–255
100	101–255
50	51–255
0	1–255

For a CoS-tagged frame, the CE-100T-8 can map the eight priorities specified in CoS to priority or best effort. The user can configure a different CoS on CTC at the card-level view under the **Provisioning > Ether Ports** tabs. Any CoS class higher than the class specified in CTC is mapped to the priority queue, which is the queue geared towards low latency. By default, the CoS is set to 7, which is the highest CoS value. This results in all traffic being treated with equal priority by default.

Table 17-2 shows which values are mapped to the priority queue for CoS settings.

Table 17-2 CoS Priority Queue Mappings

CoS Setting in CTC	CoS Values Sent to Priority Queue
7 (default)	none
6	7
5	6, 7
4	5, 6, 7
3	4, 5, 6, 7
2	3, 4, 5, 6, 7
1	2, 3, 4, 5, 6, 7
0	1, 2, 3, 4, 5, 6, 7

Ethernet frames without VLAN tagging use ToS-based priority queueing if both ToS and CoS priority queueing is active on the card. The CE-100T-8 card's ToS setting must be lower than 255 (default) and the CoS setting lower than 7 (default) for CoS and ToS priority queueing to be active. A ToS setting of 255 (default) disables ToS priority queueing, so in this case the CoS setting would be used.

Ethernet frames with VLAN tagging use CoS-based priority queueing if both ToS and CoS are active on the card. The ToS setting is ignored. CoS based priority queueing is disabled if the CoS setting is the 7 (default), so in this case the ToS setting would be used.

If the CE-100T-8 card's ToS setting is 255 (default) and the CoS setting is 7 (default), priority queueing is not active on the card, and data gets sent to the default normal traffic queue. Also if data is not tagged with a ToS value or a CoS value before it enters the CE-100T-8 card, it gets sent to the default normal traffic queue.

Note Priority queuing has no effect when flow control is enabled (default) on the CE-100T-8. Under flow control a 6 kilobyte single-priority first in first out (FIFO) buffer fills, then a PAUSE frame is sent. This results in the packet ordering priority becoming the responsibility of the external device, which is buffering as a result of receiving the PAUSE flow-control frames.

Note Priority queuing has no effect when the CE-100T-8 is provisioned with STS-3C circuits. The STS-3c circuit has more data capacity than Fast Ethernet, so CE-100T-8 buffering is not needed. Priority queuing only takes effect when buffering occurs.

RMON and SNMP Support

The CE-100T-8 card features remote monitoring (RMON) that allows network operators to monitor the health of the network with a network management system (NMS). The CE-100T-8 uses the ONG RMON. The ONG RMON contains the statistics, history, alarms, and events MIB groups from the standard RMON MIB, as well as Simple Network Management Protocol (SNMP). A user can access RMON threshold provisioning through TL1 or CTC. For RMON threshold provisioning with CTC, refer to the *Cisco ONS 15454 Procedure Guide* and the *Cisco ONS 15454 Troubleshooting Guide*. For TL1 information, refer to the *Cisco ONS SONET TL1 Command Guide*.

Statistics and Counters

The CE-100T-8 has a full range of Ethernet and POS statistics under **Performance > Ether Ports** or **Performance > POS Ports**. These are detailed in the "Performance Monitoring" chapter of the *Cisco ONS 15454 Reference Manual.*

CE-100T-8 SONET Circuits and Features

The CE-100T-8 has eight POS ports, numbered one through eight, which are exposed to management with CTC or TL1. Each POS port is statically mapped to a matching Ethernet port. By clicking the card-level Provisioning tab > POS Ports tab, the user can configure the Administrative State, Framing Type, and Encapsulation Type. By clicking the card-level Performance tab > POS Ports tab, the user can view the statistics, utilization, and history for the POS ports.

Available Circuit Sizes and Combinations

Each POS port terminates an independent contiguous SONET concatenation (CCAT) or virtual SONET concatenation (VCAT). The SONET circuit is created for these ports through CTC or TL1 in the same manner as a SONET circuit for a non-Ethernet line card. Table 17-3 shows the circuit sizes available for the CE-100T-8 on the ONS 15310-CL and ONS 15310-MA.

Table 17-3 CE-100T-8 Supported Circuit Sizes

CCAT High Order	VCAT High Order	VCAT Low Order
STS-1	STS-1-1v	VT1.5-*n*V (*n*= 1 to 64)
STS-3c	STS-1-2v	
	STS-1-3v	

A single circuit provides a maximum of 100 Mbps of throughput, even when an STS-3c circuit, which has a bandwidth equivalent of 155 Mbps, is provisioned. This is due to the hardware restriction of the Fast Ethernet port. A VCAT circuit is also restricted in this manner. Table 17-3 shows the minimum SONET circuit sizes required for 10 Mbps and 100 Mbps wire speed service.

Table 17-4 SONET Circuit Size Required for Ethernet Wire Speeds

Ethernet Wire Speed	CCAT High Order	VCAT High Order	VCAT Low Order
Line Rate 100BaseT	STS-3c	STS-1-3v, STS-1-2v*	Not applicable
Sub Rate 100BaseT	STS-1	STS-1-1v	VT1.5-*xV* (x=1-64)
Line Rate 10BaseT	STS-1	Not applicable	VT1.5-7V
Sub Rate 10BaseT	Not applicable	Not applicable	VT1.5-xV (x=1-6)

*STS-1-2v provides a total transport capacity of 98 Mbps.

The number of available circuits and total combined bandwidth for the CE-100T-8 depends on the combination of circuit sizes configured. Table 17-5 shows the circuit size combinations available for CE-100T-8 CCAT high-order circuits on the ONS 15310-CL and ONS 15310-MA. Table 17-6 shows the circuit size combinations available for CE-100T-8 VCAT high-order circuits on the ONS 15310-CL and ONS 15310-MA.

Table 17-5 CCAT High Order Circuit Size Combinations

Number of STS-3c Circuits	Maximum Number of STS-1 Circuits
None	6
1	3
2	None

Table 17-6 VCAT High Order Circuit Size Combinations

Number of STS-1-3v Circuits	Maximum Number of STS-1-2v Circuits
None	2
1	1
2	None

The CE-100T-8 supports up to eight low order VCAT circuits. The available circuit sizes are VT1.5-*n*v, where *n* ranges from 1 to 64. The total number of VT members cannot exceed 168 VT1.5s with each of the two pools on the card supporting 84 VT1.5s. The user can create a maximum of two circuits at the largest low order VCAT circuit size, VT1.5-64v.

A user can combine CCAT high order, VCAT high order, and VCAT low order circuits in any way as long as there is a maximum of eight circuits and the mapper chip bandwidth restrictions are observed. The following table details the maximum density service combinations.

Table 17-7 CE-100T-8 Maximum Service Densities

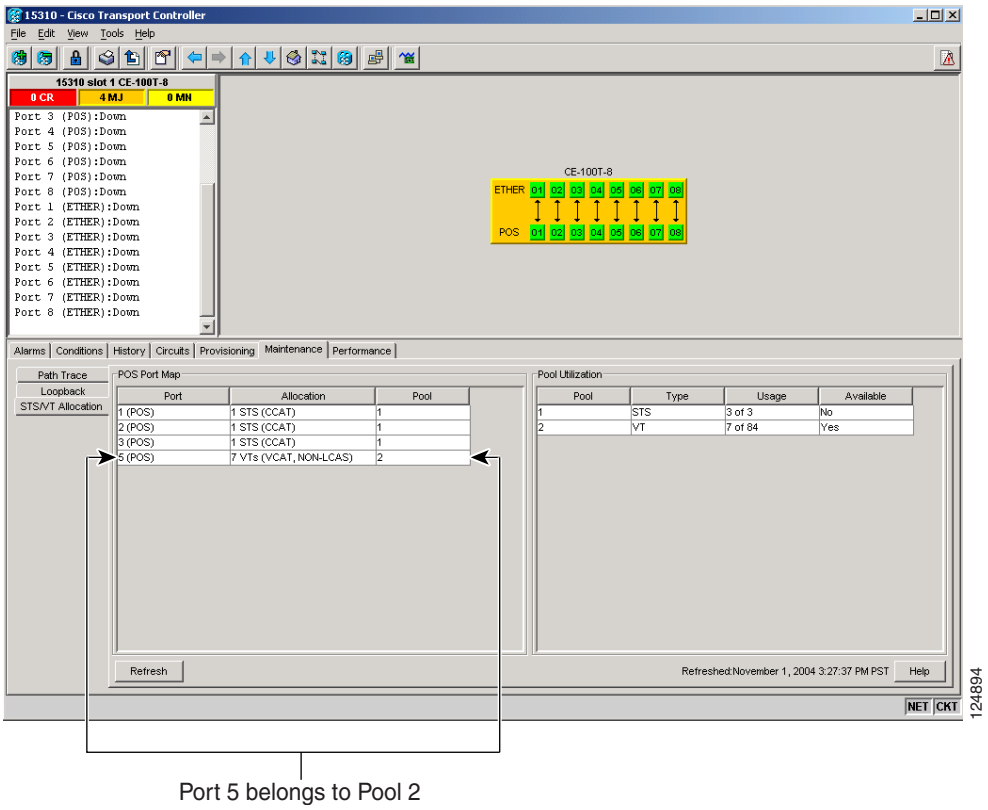
Service Combination	STS-3c or STS-1-3v	STS-1-2v	STS-1	VT1.5-xV (x=1-7)	Number of Active Service
1	2	0	0	0	2
2	1	1	1	0	3
3	1	0	3	0	4
4	1	0	0	7(x=1-12)*	8*
5	0	2	2	0	4
6	0	1	1	6(x=1-14)	8
7	0	1	0	7(x=1-12)*	8*
8	0	0	6	0	6
9	0	0	3	5(x=1-16)	8
10	0	0	0	8 (x=1-21)	8

* This LO-VCAT Circuit combination is achievable if the first circuit created on the card is an LO VCAT circuit. If the first circuit created on the card is HO-VCAT or CCAT STS circuits, then a maximum of six LO-VCAT circuits can be added on the card.

CE-100T-8 STS/VT Allocation Tab

The CE-100T-8 has two pools, each with a maximum capacity of three STSs. At the CTC card-level view under the Maintenance tab, the STS/VT Allocation tab displays how the provisioned circuits populate the two pools. This information can be useful in freeing up the bandwidth required for provisioning a circuit, if there is not enough existing capacity on any one pool for provisioning the desired circuit. The user can look at the distribution of the existing circuits among the two pools and decide which circuits to delete in order to free up space for the desired circuit.

Figure 17-4 CE-100T-8 STS/VT Allocation Tab



For example if a user needs to provision an STS-3c or STS-1-3v on the CE-100T-8 card shown in Figure 17-4, an STS-3c or STS-1-3v worth of bandwidth is not available from either of the two pools. The user needs to delete circuits from the same pool to free up bandwidth. If the bandwidth is available but scattered among the pools, the circuit cannot be provisioned.

Looking at the POS Port Map table, the user can determine which circuits belong to which pools. The Pool and Port columns in Figure 17-4 show that the circuit on port 5 is drawn from Pool 2, and no other circuits are drawn from Pool 2. Deleting this one circuit will free up an STS-3c or STS-1-3v worth of bandwidth from a single pool.

The POS Port table has a row for each port with three columns (Figure 17-4). They show the port number, the circuit size and type, and the pool it is drawn from. The Pool Utilization table has two columns and shows the pool number, the type of circuits on that pool, how much of the pool's capacity is being used, and whether additional capacity is available.

CE-100T-8 VCAT Characteristics

The ML-100T-8 card and the CE-100T-8 card (both the version for the ONS 15310-CL and ONS 15310-MA and the version for the ONS 15454 SONET/SDH) have hardware-based support for the ITU-T G.7042 standard link capacity adjustment scheme (LCAS). This allows the user to dynamically resize a high order or low order VCAT circuit through CTC or TL1 without affecting other members of the VCG (errorless). ML-100T-8 LCAS support is high order only and is limited to a two member VCG.

To enable end-to-end connectivity in a VCAT circuit that traverses through a third-party network, you must create a server trail between the ports. For more details, refer to the "Create Circuits and VT Tunnels" chapter in the *Cisco ONS 15310-CL and Cisco ONS 15310-MA Procedure Guide*.

The ONS 15454 SONET/SDH ML-Series card has a software-based LCAS (SW-LCAS) scheme. This scheme is also supported by both the ML-100T-8 card and both versions of the CE-100T-8, but only for circuits with the other end terminating on an ONS 15454 SONET/SDH ML-Series card.

The CE-100T-8 card allows independent routing and protection preferences for each member of a VCAT circuit. The user can also control the amount of VCAT circuit capacity that is fully protected, unprotected or if the circuit is on a bidirectional line switched ring (BLSR), uses protection channel access (PCA). Alarms are supported on a per-member as well as per virtual concatenation group (VCG) basis.

Note The maximum tolerable VCAT differential delay for the CE-100T-8 is 48 milliseconds. The VCAT differential delay is the relative arrival time measurement between members of a virtual concatenation group (VCG).

On ML-100T-8 and CE-100T-8 cards, members of a HW-LCAS circuit must be moved to the OOS,OOG (locked, outOfGroup) state before you delete them.

A traffic hit is seen under the following conditions:

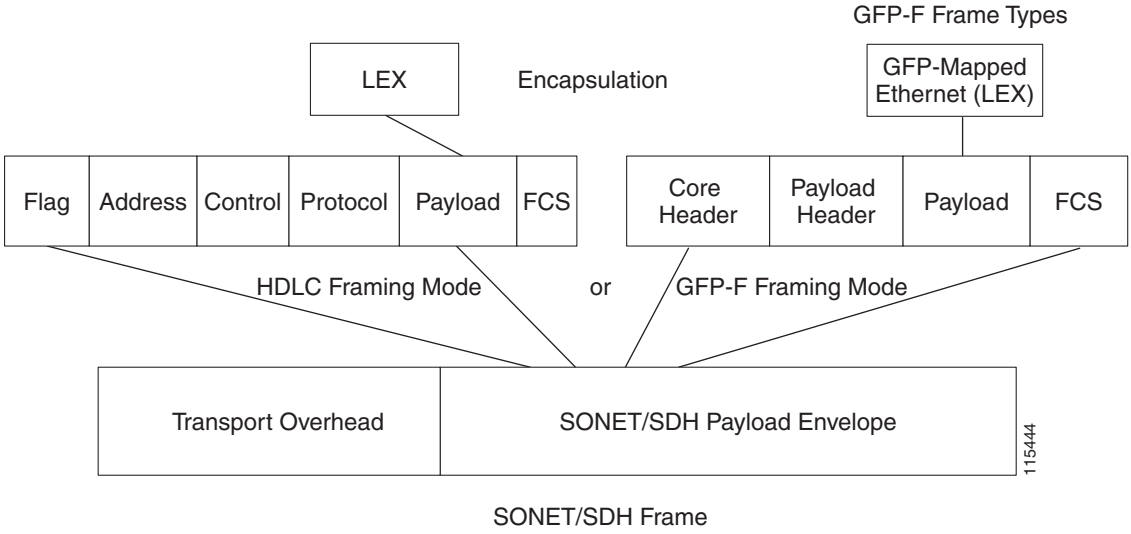
- A hard reset of the card containing the trunk port.
- Trunk port moved to OOS,DSBLD (locked,disabled) state.
- Trunk fiber pull.
- Deletion of members of the HW-LCAS circuit in IG (In Group) state.

CE-100T-8 POS Encapsulation, Framing, and CRC

The CE-100T-8 uses Cisco EoS LEX (LEX). LEX is the primary encapsulation of ONS Ethernet cards. In this encapsulation the protocol field is set to the values specified in Internet Engineering Task Force (IETF) Request For Comments (RFC) 1841. The user can provision GPF-F framing (default) or high-level data link control (HDLC) framing. With GFP-F framing, the user can also configure a 32-bit CRC (the default) or no CRC (none). aming. With GFP-F framing, the user can also configure a 32-bit CRC (the default) or no CRC (none). On CTC go to CE card view and click the Provisioning >pos ports tab, to see the various parameters that can be configured on the POS ports, see Displaying ML-Series Ethernet Statistics in CTC, page 2-2. Various parameters like, admin state, service state, framing type, CRC , MTU and soak time for a port can be configured hereWhen LEX is used over GFP-F it is standard Mapped Ethernet over GFP-F according to ITU-T G.7041. HDLC framing provides a set 32-bit CRC.

Figure 17-5 illustrates CE-100T-8 framing and encapsulation.

Figure 17-5 ONS CE-100T-8 Encapsulation and Framing Options



The CE-100T-8 card supports GFP-F null mode. GFP-F CMFs are counted and discarded.

The CE-100T-8 card is interoperable with the ML-100T-8 card and several other ONS Ethernet cards. For specific details on ONS Ethernet card interoperability, refer to the "POS on ONS Ethernet Cards" chapter of the *Cisco ONS 15454 and Cisco ONS 15454 SDH Ethernet Card Software Feature and Configuration Guide*.

CE-100T-8 Loopback, J1 Path Trace, and SONET Alarms

The CE-100T-8 card supports terminal and facility loopbacks when in the Out of Service, Maintenance state (OOS, MT). It also reports SONET alarms and transmits and monitors the J1 Path Trace byte in the same manner as OC-N cards. Support for path termination functions includes:

- H1 and H2 concatenation indication
- C2 signal label
- Bit interleaved parity 3 (BIP-3) generation
- G1 path status indication
- C2 path signal label read/write
- Path level alarms and conditions, including loss of pointer, unequipped, payload mismatch, alarm indication signal (AIS) detection, and remote defect indication (RDI)
- J1 path trace for high order paths
- J2 path trace for low order paths
- J2 path trace for low order VCAT circuits at the member level
- Extended signal label for the low order paths

APPENDIX A

Command Reference for the ML-Series Card

Note The terms "Unidirectional Path Switched Ring" and "UPSR" may appear in Cisco literature. These terms do not refer to using Cisco ONS 15xxx products in a unidirectional path switched ring configuration. Rather, these terms, as well as "Path Protected Mesh Network" and "PPMN," refer generally to Cisco's path protection feature, which may be used in any topological network configuration. Cisco does not recommend using its path protection feature in any particular topological network configuration.

This appendix provides a command reference for those Cisco IOS commands or those aspects of Cisco IOS commands that are unique to ML-Series cards. For information about the standard Cisco IOS Release 12.2 commands, refer to the Cisco IOS documentation set available at http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/.

[no] bridge *bridge-group-number* protocol {drpri-rstp | ieee | rstp}

To define the protocol employed by a bridge group, use the **bridge protocol** global configuration command. If no protocol will be employed by the bridge group, this command is not needed. To remove a protocol from the bridge group, use the no form of this command with the appropriate keywords and arguments.

Syntax Description

Parameter	Description
drpri-rstp	The protocol that enables the Dual Resilient Packet Ring Interconnect (DRPRI) feature of the ONS 15454 ML-Series cards. Do not configure an ONS 15310-CL or ONS 15310-MA ML-Series card with this option.
ieee	IEEE 802.1D Spanning Tree Protocol (RSTP).
rstp	IEEE 802.1W Rapid Spanning Tree Protocol (STP).
bridge-group-number	The identifying number of the bridge group being assigned a protocol.

Defaults

N/A

Command Modes

Global configuration

Usage Guidelines

The ONS 15310-CL or ONS 15310-MA ML-Series card implement RSTP or STP. DRPRI is not available.

Examples

The following example assigns STP to the bridge group with the bridge group number of 100.

```
Router(config)# bridge 100 protocol ieee
```

Related Commands

bridge-group

clear counters

Use the **clear counters** command to simultaneously clear Ethernet interface performance monitoring (PM) counters in Cisco Transport Controller (CTC), Transaction Language One (TL1), and the Cisco IOS CLI. Using Cisco IOS, you can clear counters on a per-interface basis for any except the 802.13 IEEE RPR interface; in that instance, you can only clear all counters for both spans.

The clear command can also be executed from CTC by means of a button, or from TL1 using a command on the interface. The CTC clearing function allows you to choose between clearing front-end or back-end interfaces. Cisco IOS and TL1 interface clear commands do not have this ability.

Syntax Description

This command has no arguments or keywords.

Defaults

The default is for PM counters not to be cleared.

Command Modes

Privileged exec

Usage Guidelines

This command is applicable to the ML100T-8 card on the ONS 15310-CL and ONS 15310-MA.

Examples

```
Router# clear counters
Clear "show interface" counters on all interfaces [confirm]
Router#
```

Related Commands

show interface

[no] clock auto

Use the **clock auto** command to determine whether the system clock parameters are configured automatically from the node's timing. When enabled, both daylight savings time and time zone are automatically configured, and the system clock is periodically synchronized to the node's timing. Use the no form of the command to disable this feature.

Syntax Description

This command has no arguments or keywords.

Defaults

The default setting is clock auto.

Command Modes

Global configuration

Usage Guidelines

The no form of the command is required before any manual configuration of daylight savings time, time zone, or clock. The no form of the command is required if Network Time Protocol (NTP) is configured in Cisco IOS. The ONS 15310-CL and ONS 15310-MA are also configured through Cisco Transport Controller (CTC) to use an NTP or Simple Network Time Protocol (SNTP) server to set the date and time of the node.

Examples

```
ML_Series(config)# no clock auto
```

Related Commands

clock summertime

clock timezone

clock set

interface spr 1

Use this command to create a shared packet ring (SPR) interface on an ML-Series card for a resilient packet ring (RPR). If the interface has already been created, this command enters spr interface configuration mode. The only valid spr interface number is 1.

Defaults N/A

Command Modes Global configuration

Usage Guidelines The command allows the user to create a virtual interface for the RPR/SPR. Commands such as **spr wrap** or **spr station-id** can then be applied to the RPR through SPR configuration command mode.

Examples The following example creates the shared packet ring interface:

```
ML_Series(config)# interface spr 1
```

Related Commands spr-intf-id

spr station-id

spr wrap

[no] pos mode gfp [fcs-disabled]

Sets the framing mode employed by the ONS Ethernet card for framing and encapsulating data packets onto the SONET transport layer. Valid framing modes are:

- HDLC—(High-level data link control) A common mechanism employed in framing data packets for SONET/SDH.
- GFP (default)—The ML-Series card supports the frame mapped version of generic framing procedure (GFP-F).

Note The GFP-F FCS is compliant with ITU-T G.7041/Y.1303.

Defaults

The default framing mode is GFP-F with a 32-bit frame check sequence (FCS) enabled.

Syntax Description

The optional fcs-disabled keyword disables the GFP-F FCS. The no form of the command sets the framing mode to Cisco HDLC. The fcs-disabled keyword is not available when setting the framing mode to Cisco HDLC.

Command Modes

Interface configuration mode (Packet-over-SONET [POS] only)

Usage Guidelines

This command can be used only when the ML-Series card's POS interface is in shutdown mode. The peer path terminating element (PTE) needs to be in the same framing mode as the POS interface.

Examples

```
ML_Series(config) # int pos0
ML_Series(config-if) # shutdown
ML_Series(config-if) # pos mode gfp fcs-disable
ML_Series(config-if) # no shutdown
```

Related Commands

shutdown

[no] pos pdi holdoff *time*

Use this command to specify the time, in milliseconds, to hold off sending the path defect indication (PDI) to the far-end when a VCAT member circuit is added to the virtual concatenation group (VCG). Use the no form of the command to use the default value.

Syntax Description

Parameter	Description
time	delay time in milliseconds, 100 to 1000

Defaults

The default value is 100 milliseconds.

Command Modes

Interface configuration mode (POS only)

Usage Guidelines

This value is normally configured to match the setting on the peer PTE. The time granularity for this command is 1 millisecond.

Examples

```
Gateway(config)# int pos0
Gateway(config-if)# pos pdi holdoff 500
```

Related Commands

pos trigger defects

[no] pos report *alarm*

Use this command to specify which alarms/signals are logged to the console. This command has no effect on whether alarms are reported to the TCC2/TCC2P and CTC. These conditions are soaked and cleared per Telcordia GR-253. Use the no form of the command to disable reporting of a specific alarm/signal.

Syntax Description

Parameter	Description
alarm	The SONET/SDH alarm that is logged to the console. The alarms are as follows: **all**—All link down alarm failures **ber_sd_b3**—PBIP BER in excess of SD threshold failure **ber_sf_b3**—PBIP BER in excess of SF threshold failure **encap**—Path signal label encapsulation mismatch failure **pais**—Path alarm indication signal failure **plop**—Path loss of pointer failure **ppdi**—Path payload defect indication failure **pplm**—Payload label mismatch path **prdi**—Path remote defect indication failure **ptim**—Path trace indicator mismatch failure **puneq**—Path label equivalent to zero failure

Defaults

The default is to report all alarms.

Command Modes

Interface configuration mode (POS only)

Usage Guidelines

This value is normally configured to match the setting on the peer PTE.

Examples

```
Gateway(config)# int pos0

Gateway(config-if)# pos report all

Gateway(config-if)# pos flag c2 1
03:16:51: %SONET-4-ALARM:  POS0: PPLM

Gateway(config-if)# pos flag c2 0x16
03:17:34: %SONET-4-ALARM:  POS0: PPLM  cleared
```

Related Commands

pos trigger defects

[non] pos trigger defects *condition*

Use this command to specify which conditions cause the associated POS link state to change. These conditions are soaked/cleared using the delay specified in the **pos trigger delay** command. Use the no form of the command to disable triggering on a specific condition.

Syntax Description

Parameter	Description
condition	The SONET/SDH condition that causes the link state change. The conditions are as follows: **all**—All link down alarm failures (default) **ber_sd_b3**—PBIP BER in excess of SD threshold failure **ber_sf_b3**—PBIP BER in excess of SF threshold failure **encap**—Path signal label encapsulation mismatch failure **pais**—Path alarm indication signal failure **plop**—Path loss of pointer failure **ppdi**—Path payload defect indication failure **pplm**—Payload label mismatch path **prdi**—Path remote defect indication failure **ptim**—Path trace indicator mismatch failure **puneq**—Path label equivalent to zero failure

Defaults

The default is to report **all** conditions except **ber_sd_b3**. For a list of all conditions, see the list in the above description.

Command Modes

Interface configuration mode (POS only)

Usage Guidelines

This value is normally configured to match the setting on the peer PTE.

Examples

```
Gateway(config)# int pos0
Gateway(config-if)# pos trigger defects all
```

Related Commands

pos trigger delay

[no] pos trigger delay *time*

Use this command to specify which conditions cause the associated POS link state to change. The conditions specified in the **pos trigger defects** command are soaked/cleared using this delay. Use the no form of the command to use the default value.

Syntax Description

Parameter	Description
time	delay time in milliseconds, 200 to 2000

Defaults

The default value is 200 milliseconds.

Command Modes

Interface configuration mode (POS only)

Usage Guidelines

This value is normally configured to match the setting on the peer PTE. The time granularity for this command is 50 milliseconds.

Examples

```
Gateway(config)# int pos0
Gateway(config-if)# pos trigger delay 500
```

Related Commands

pos trigger defects

[no] pos vcat defect {immediate | delayed}

Sets the virtual concatenated (VCAT) defect processing mode to either handle a defect state change the instant it is detected or wait for the time specified by **pos trigger delay.** Use the no form of the command to use the default value.

Syntax Description

Parameter	Description
immediate	Handles a defect state change the instant it is detected.
delayed	Handles the defect after the time specified by the command pos trigger delay. If delay is configured and the circuit is on RPR, then the RPR defect processing will also be delayed by the delay time.

Defaults

The default setting is immediate.

Command Modes

POS interface configuration

Usage Guidelines

Immediate should be used if the VCAT circuit uses unprotected SONET circuits. Delayed should be run if the VCAT circuit uses SONET protected circuits, such as apath protection.

Examples

The following example sets an ML-Series card to delayed:

```
ML_Series(config)# interface pos 1
ML_Series(config-if)# pos vcat defect delayed
```

Related Commands

interface spr 1

spr wrap

interface pos 1

pos trigger delay

show controller pos *interface-number* [details]

Use this command to display the status of the POS controller. Use the details argument to obtain certain additional information.

Syntax Description

Parameter	Description
interface-number	Number of the POS interface (0–1)

Defaults

N/A

Command Modes

Privileged EXEC

Usage Guidelines

This command can be used to help diagnose and isolate POS or SONET problems.

Examples

The following example displays the ML-Series controller information for interface pos 0.

```
ML_Series# show controller pos 0
Interface POS0
Hardware is Packet Over SONET
Framing Mode: HDLC
Concatenation: CCAT
Alarms reportable to CLI: AIS-P LOP-P UNEQ-P TIM-P PLM-P ENCAP-MISMATCH RDI-P PDI-P SF-P
SD-P
Link state change defects: AIS LOP UNEQ TIM PLM ENCAP RDI PDI
Link state change time   : 200 (msec)
***************  Path  ***************
Circuit Type: STS-1
Physical Channel Number: 0
Circuit ESM State: IS
STS Index 0
Active Alarms: None
B3 BER thresholds:
SFBER = 1e-4,   SDBER = 1e-7
    Path Trace Info.
Channel 0
  Received String Format : 64 Byte
  Transmit String Format : 64 Byte
  Provisioned Trace Mode : off
  Prov'd   : false   TIU-P   : FALSE    TIM-P : FALSE
  State    : w4xcon      MatchCnt: 0  MisMatchCnt: 0
  Rec Flag : false    Exp Flag : false  Xmt Enab : true
2398983617 total input packets,  1913918056382 post-decap bytes
0 input short packets
67757  input CRCerror packets , 0  input drop packets
63584  rx HDLC addr mismatchs , 63599  rx HDLC ctrl mismatchs
63630  rx HDLC sapi mismatchs , 63599  rx HDLC ctrl mismatchs
289  rx HDLC destuff errors , 68048  rx HDLC invalid frames
0 input abort packets
2093 input packets dropped by ucode
0 input packets congestion events
2398847783 input good packets (POS MAC tx)
```

```
1913918056382 input good octets (POS MAC tx)
2397888202 total output packets,  1913918056382 pre-encap bytes
Carrier delay is 200 msec
```

Related Commands show interface pos

clear counters

show interface pos *interface-number*

Use this command to display the status of the POS interface.

Syntax Description

Parameter	Description
interface-number	Number of the POS interface (0–1)

Defaults

N/A

Command Modes

Privileged EXEC

Usage Guidelines

This command can be used to help diagnose and isolate POS or SONET problems.

Examples

The following example displays the ML-Series interface information for interface pos 0.

```
ML_Series# show interfaces pos0
POS0 is up, line protocol is up
  Hardware is Packet Over SONET, address is 000c.9a9a.9a9a (bia 000c.9a9a.9a9a)
  MTU 1500 bytes, BW 48384 Kbit, DLY 100 usec,
     reliability 255/255, txload 157/255, rxload 157/255
 Encapsulation: Cisco-EoS-LEX, loopback not set
  Keepalive set (10 sec)
  Scramble enabled
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:00, output never, output hang never
  Last clearing of "show interface" counters 5d22h
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 29797000 bits/sec, 4673 packets/sec
  5 minute output rate 29841000 bits/sec, 4670 packets/sec
     2399801434 packets input, 3309269642 bytes
     Received 799619391 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
              0 parity
     135834 input errors, 67757 CRC, 0 frame, 0 overrun, 0 ignored
     0 input packets with dribble condition detected
     2398705102 packets output, 1211912638 bytes, 0 underruns
     0 output errors, 0 applique, 0 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions
```

Related Commands

show controller pos

clear counters

show ons alarm

Use this command to display all the active alarms on the card.

Syntax Description This command has no arguments or keywords.

Defaults N/A

Command Modes Privileged EXEC

Usage Guidelines This command can be used to help diagnose and isolate card problems.

Examples

```
ML_Series# show ons alarm
Equipment Alarms
Active: CONTBUS-IO-A CTNEQPT-PBWORK

Port Alarms
  POS0  Active: None
  POS1  Active: None
  FastEthernet0  Active: None
  FastEthernet1  Active: None
  FastEthernet2  Active: None
  FastEthernet3  Active: None
  FastEthernet4  Active: None
  FastEthernet5  Active: None
  FastEthernet6  Active: None
  FastEthernet7  Active: None
  FastEthernet8  Active: None
  FastEthernet9  Active: None
  FastEthernet10  Active: None
  FastEthernet11  Active: None

POS0

Active Alarms : None
Demoted Alarms: None

POS1 VCG State: VCG_NORMAL
VCAT Group
Active Alarms : None
Demoted Alarms: None

Member 0
Active Alarms : None
Demoted Alarms: None

Member 1
Active Alarms : None
Demoted Alarms: None
```

Related Commands

show controller pos
show ons alarm defects
show ons alarm failures

show ons alarm defect {[eqpt | port [*port-number*] | sts [*sts-number*] | vcg [*vcg-number*] | vt]}

This command displays all defects for the ML-Series card with no keyword (default) or defects for the level specified by the keyword.

Syntax Description

Parameter	Description
eqpt	Specifies hardware-related.
port	Specifies the physical interface level. Optional *port-number* specifies a particular physical interface.
sts	Specifies the SONET circuit level. Optional *sts-number* specifies a particular SONET circuit.
vcg	Specifies the VCAT circuit group level. Optional *vcg-number* specifies a particular VCAT group.
vt	Not valid.

Defaults

Displays all defects

Command Modes

Privileged EXEC

Usage Guidelines

This command displays the set of active defects for the specified layer and the possible set of defects that can be set.

Examples

The following example shows the command and output for the ML-Series alarm defect information at the equipment level.

```
ML_Series# show ons alarm defect eqpt
Equipment Defects
Active: RUNCFG-SAVENEED
Reportable to SC/CLI: CONTBUS-IO-A CONTBUS-IO-B CTNEQPT-PBWORK CTNEQPT-PBPROT EQPT
RUNCFG-SAVENEED ERROR-CONFIG HIGH-TEMP PROVISION-ERROR
Port Defects
  POS0
  Active: None
  Reportable to SC: CARLOSS TPTFAIL GFP-LFD GFP-CSF GFP-UPI LPBK-TERMINAL LPBK-FACILITY
  POS1
........
```

Note The output example is abbreviated because of length.

The following example shows the command and output for the ML-Series alarm defect information at the port level.

```
ML-Series# show ons alarm defect port
```

```
Port Defects
  POS0
  Active: None
  Reportable to SC: CARLOSS TPTFAIL GFP-LFD GFP-CSF GFP-UPI LPBK-TERMINAL LPBK-FACILITY
  POS1
  Active: None
  Reportable to SC: CARLOSS TPTFAIL GFP-LFD GFP-CSF GFP-UPI LPBK-TERMINAL LPBK-FACILITY
  FastEthernet0
  Active: None
  Reportable to SC: CARLOSS TPTFAIL GFP-LFD GFP-CSF GFP-UPI LPBK-TERMINAL LPBK-FACILITY
  FastEthernet1
  Active: None
  Reportable to SC: CARLOSS TPTFAIL GFP-LFD GFP-CSF GFP-UPI LPBK-TERMINAL LPBK-FACILITY
  FastEthernet2
  Active: None
  Reportable to SC: CARLOSS TPTFAIL GFP-LFD GFP-CSF GFP-UPI LPBK-TERMINAL LPBK-FACILITY
  FastEthernet3
  Active: None
  Reportable to SC: CARLOSS TPTFAIL GFP-LFD GFP-CSF GFP-UPI LPBK-TERMINAL LPBK-FACILITY
  FastEthernet4
  Active: None
  Reportable to SC: CARLOSS TPTFAIL GFP-LFD GFP-CSF GFP-UPI LPBK-TERMINAL LPBK-FACILITY
  FastEthernet5
  Active: None
  Reportable to SC: CARLOSS TPTFAIL GFP-LFD GFP-CSF GFP-UPI LPBK-TERMINAL LPBK-FACILITY
  FastEthernet6
  Active: None
  Reportable to SC: CARLOSS TPTFAIL GFP-LFD GFP-CSF GFP-UPI LPBK-TERMINAL LPBK-FACILITY
  FastEthernet7
  Active: None
  Reportable to SC: CARLOSS TPTFAIL GFP-LFD GFP-CSF GFP-UPI LPBK-TERMINAL LPBK-FACILITY
```

The following example shows the command and output for the ML-Series alarm defect information at the synchronous transport signal (STS) level.

```
ML_Series# show ons alarm defect sts
STS Defects
  STS 0
  Active: None
  STS 1
  Active: None
  STS 2
  Active: None
  STS 3
  Active: None
  STS 4
  Active: None
  STS 5
  Active: None
```

Related Commands show controller pos

show ons alarm failures

show ons alarm failure {[eqpt | port [*port-number*] | sts [*sts-number*] | vcg [*vcg-number*] | vt]}

This command displays all failures for the ML-Series card with no keyword (default) or failures for the level specified by the keyword.

Syntax Description

Parameter	Description
eqpt	Specifies hardware-related.
port	Specifies the physical interface level. Optional *port-number* specifies a particular physical interface.
sts	Specifies the SONET circuit level. Optional *sts-number* specifies a particular SONET circuit.
vcg	Specifies the VCAT circuit group level. Optional *vcg-number* specifies a particular VCAT group.
vt	Not valid.

Defaults

N/A

Command Modes

Privileged EXEC

Usage Guidelines

This command displays the set of active failures for the specified layer and the possible set of failures that can be set.

Examples

The following example shows the command and output for the ML-Series alarm failure information at the equipment level.

```
ML_Series# show ons alarm failure eqpt
Equipment Alarms
Active: RUNCFG-SAVENEED
```

The following example shows the command and output for the ML-Series alarm failure information at the port level.

```
ML-Series# show ons alarm failure port
Port Alarms
  POS0  Active: None
  POS1  Active: None
  FastEthernet0  Active: None
  FastEthernet1  Active: None
  FastEthernet2  Active: None
  FastEthernet3  Active: None
  FastEthernet4  Active: None
  FastEthernet5  Active: None
  FastEthernet6  Active: None
  FastEthernet7  Active: None
```

The following example shows the command and output for the ML-Series alarm failure information at the STS level.

```
ML_Series# show ons alarm failure sts
STS Defects
  STS 0
  Active: None
  STS 1
  Active: None
  STS 2
  Active: None
  STS 3
  Active: None
  STS 4
  Active: None
  STS 5
  Active: None
```

Related Commands show ons alarm defect

show interface

show controller pos

spr-intf-id *shared-packet-ring-number*

Assigns the POS interface to the SPR interface.

Syntax Description

Parameter	Description
shared-packet-ring-number	The only valid shared-packet-ring-number (SPR number) is 1.

Defaults

N/A

Command Modes

POS interface configuration

Usage Guidelines

- The SPR number must be 1, which is the same SPR number assigned to the SPR interface.
- The members of the SPR interface must be POS interfaces.
- An SPR interface is configured similarly to a EtherChannel (port-channel) interface. Instead of using the **channel-group** command to define the members, you use the **spr-intf-ID** command. Like port-channel, you then configure the SPR interfaces instead of the POS interface.

Examples

The following example assigns an ML-Series card POS interface to an SPR interface with a shared-packet-ring-number of 1:

```
ML_Series(config)# interface pos 0
ML_Series(config-if)# spr-intf-id 1
```

Related Commands

interface spr 1

spr station-id

spr wrap

[no] spr load-balance { auto | port-based }

Specifies the RPR load-balancing scheme for unicast packets.

Syntax Description

Parameter	Description
auto	The default auto option balances the load based on the MAC addresses or the source and destination addresses of the IP packet.
port-based	The port-based load balancing option maps unicast packets from even ports to the POS 0 interface and odd ports to the POS 1 interface.

Defaults

The default setting is auto.

Command Modes

SPR interface configuration

Examples

The following example configures an SPR interface to use port-based load balancing:

```
ML_Series(config)# interface spr 1
ML_Series(config-if)# spr load-balance port-based
```

Related Commands

interface spr 1

spr station-id *station-id-number*

Configures a station ID.

Syntax Description

Parameter	Description
station-id-number	The user must configure a different number for each SPR interface that attaches to the RPR. Valid station ID numbers range from 1 to 254.

Defaults

N/A

Command Modes

SPR interface configuration

Usage Guidelines

The different ML-Series cards attached to the RPR all have the same interface type and number, spr1. The station ID helps to differentiate the SPR interfaces.

Examples

The following example sets an ML-Series card SPR station ID to 100:

```
ML_Series(config)# interface spr 1
ML_Series(config-if)# spr station-id 100
```

Related Commands

interface spr 1

spr-intf-id

spr wrap

spr wrap { immediate | delayed }

Sets the RPR wrap mode to either wrap traffic the instant it detects a link state change or to wrap traffic after the carrier delay, which gives the SONET protection time to register the defect and declare the link down.

Syntax Description

Parameter	Description
immediate	Wraps RPR traffic the instant it detects a link state change.
delayed	Wraps RPR traffic after the carrier delay time expires.

Defaults

The default setting is immediate.

Command Modes

SPR interface configuration

Usage Guidelines

Immediate should be used if RPR is running over unprotected SONET circuits. Delayed should be run for SONET protected circuits (bidirectional line switched ring [BLSR] or path protection.

Examples

The following example sets an ML-Series card to delayed:

```
ML_Series(config)# interface spr 1
ML_Series(config-if)# spr wrap delayed
```

Related Commands

interface spr 1

spr-intf-id

spr station-id

APPENDIX **B**

Unsupported CLI Commands for the ML-Series Card

This appendix lists some of the command-line interface (CLI) commands that are not supported in this release, either because they were not tested, or because of hardware limitations. These unsupported commands are displayed when you enter the question mark (?) at the CLI prompt. This is not a complete list. Unsupported commands are listed by command mode.

Unsupported Privileged Exec Commands

clear ip accounting

show controller pos pm

show controller pos [*variable*] pm

show ip accounting

show ip cache

show ip tcp header-compression

show ip mcache

show ip mpacket

show ons alarm defect vt

show ons alarm failure vt

Unsupported Global Configuration Commands

access-list aaa <1100-1199>

access-list aaa <200-299>

access-list aaa <700-799>

async-bootp

boot

bridge <num> acquire

bridge <num> address

bridge cmf

bridge <num> bitswap-layer3-addresses

bridge <num> circuit-group

bridge <num> domain

bridge <num> lat-service-filtering

bridge <num> protocol dec

bridge <num> protocol ibm

bridge <num> protocol vlan-bridge

chat-script

class-map match access-group

class-map match class-map

class-map match destination-address

class-map match mpls

class-map match protocol

class-map match qos-group

class-map match source-address

clns

define

dialer

dialer-list

downward-compatible-config

file

ip access-list log-update

ip access-list logging

ip address-pool

ip alias

ip bootp

ip gdp

ip local

ip radius nas-ip-address (Command distinguishes multiple ONS 15454 SONET/SDH ML-Series cards.)

ip reflexive-list

ip security

ip source-route

ip tcp

ipc

map-class

map-list

multilink

netbios

partition

policy-map class queue-limit

priority-list

queue-list

router iso-igrp

router mobile

service compress-config

service disable-ip-fast-frag

service exec-callback

service nagle

service old-slip-prompts

service pad

service slave-log

subscriber-policy

Unsupported POS Interface Configuration Commands

access-expression

autodetect

bridge-group x circuit-group

bridge-group x input-*

bridge-group x lat-compression

bridge-group x output-*

bridge-group x subscriber-loop-control

clock

clns

crc 32 (CRC is 32-bit size by default and cannot be configured on the ONS 15310-CL and ONS 15310-MA ML-Series card.)

custom-queue-list

down-when-looped

fair-queue

flowcontrol

full-duplex

half-duplex

hold-queue

ip accounting

ip broadcast-address

ip load-sharing per-packet
ip route-cache
ip security
ip tcp
ip verify
iso-igrp
loopback
multilink-group
netbios
pos flag c2
pos scramble-spe
pos vcat resequence
priority-group
pulse-time
random-detect
rate-limit
rmon
scramble
serial
service-policy history
source
timeout
transmit-interface
tx-ring-limit

Unsupported FastEthernet Interface Configuration Commands

access-expression
clns
custom-queue-list
fair-queue
hold-queue
ip accounting
ip broadcast-address
ip load-sharing per-packet
ip route-cache
ip security
ip tcp

ip verify
iso-igrp
keepalive
loopback
max-reserved-bandwidth
multilink-group
netbios
priority-group
random-detect
rate-limit
service-policy history
timeout
transmit-interface
tx-ring-limit

Unsupported Port-Channel Interface Configuration Commands

access-expression
carrier-delay
cdp
clns
custom-queue-list
duplex
down-when-looped
encapsulation
fair-queue
flowcontrol
full-duplex
half-duplex
hold-queue
iso-igrp
keepalive
max-reserved-bandwidth
multilink-group
negotiation
netbios
ppp
priority-group

rate-limit

random-detect

timeout

tx-ring-limit

Unsupported BVI Interface Configuration Commands

access-expression

carrier-delay

cdp

clns

flowcontrol

hold-queue

iso-igrp

keepalive

l2protocol-tunnel

load-interval

max-reserved-bandwidth

mode

multilink-group

netbios

ntp

mtu

rate-limit

timeout

transmit-interface

tx-ring-limit

APPENDIX C

Using Technical Support

This appendix describes how to resolve problems with your ML-Series card and contains the following sections:

To help resolve these problems, use the "Gathering Information About Your Internetwork" section on page C-1 as a guideline for gathering relevant information about your network prior to calling.

Note When you have a problem that you cannot resolve, contact the Cisco Technical Assistance Center (TAC). See the "Obtaining Documentation, Obtaining Support, and Security Guidelines" section on page ix as needed.

Gathering Information About Your Internetwork

Before gathering any specific data, compile a list of all symptoms that users have reported on the internetwork (such as connections dropping or slow host response).

The next step is to gather specific information. Typical information needed to troubleshoot internetworking problems falls into two general categories: information required for any situation; and information specific to the topology, technology, or protocol.

Information that is always required by technical support engineers includes the following:

- Network topology map for the data network and the SONET topology and provisioning.
- List of hosts and servers: Include the host and server type, number on network, and a description of the host operating systems that are implemented.
- Configuration listing of all switch routers and switches involved.
- Complete specifications of all switch routers and switches involved.
- Version numbers of software (obtained with the **show version** command) and flash code (obtained with the **show controllers** command) on all relevant switch routers and switches.
- List of network layer protocols, versions, and vendors.
- List of alarms and conditions on all nodes in the SONET/SDH topology.
- Node equipment and configuration.

To assist you in gathering this required data, the **show tech-support** EXEC command has been added in Cisco IOS Release 11.1(4) and later. This command provides general information about the switch router that you can provide to your technical support representative when you are reporting a problem.

The **show tech-support** command outputs the equivalent of the **show version**, **show running-config**, **show controllers**, **show stacks**, **show interfaces**, **show buffers**, **show process memory**, and **show process** EXEC commands.

The specific information requirements that might be needed by technical support vary depending on the situation. They include the following:

- Output from the following general **show** commands:

  show interfaces

  show controllers

  show processes {**cpu** | **mem**}

  show buffer

  show mem summary
- Output from the following protocol-specific **show** commands:

  show *protocol* **route**

  show *protocol* **traffic**

  show *protocol* **interfaces**

  show *protocol* **arp**
- Output from provisioning show commands
- Output from relevant **debug** privileged EXEC commands
- Output from protocol-specific **ping** and **trace** diagnostic tests, as appropriate
- Network analyzer traces, as appropriate
- Core dumps obtained using the **exception dump** command, or using the **write core** command if the system is operational, as appropriate

Getting the Data from Your ML-Series Card

When obtaining the information from your ML-Series card, you must tailor your method to the system that you are using to retrieve the information. Following are some hints for different platforms:

- PC and Macintosh—Connect a PC or Macintosh to the console port of the ML-Series card and log all output to a disk file (using a terminal emulation program). The exact procedure varies depending on the communication package used with the system.
- Terminal connected to the console port or remote terminal—The only way to get information with a terminal connected to the console port or with a remote terminal is to attach a printer to the AUX port on the terminal (if one exists) and to force all screen output to go to the printer. Using a terminal is undesirable because there is no way to capture the data to a file.
- UNIX workstation—At the UNIX prompt, enter the command **script** *filename*, then use Telnet to connect to the ML-Series card. The UNIX **script** command captures all screen output to the specified filename. To stop capturing output and close the file, enter the end-of-file character (typically **Ctrl-D**) for your UNIX system.

Note To get your system to automatically log specific error messages or operational information to a UNIX syslog server, enter the **logging** *internet-address* command. For more information about using the **logging** command and setting up a syslog server, refer to the Cisco IOS configuration guides and command references.

Providing Data to Your Technical Support Representative

When submitting information to your technical support representative, electronic data is preferred. Electronic data significantly eases the transfer of information between technical support personnel and development staff. Common electronic formats include data sent through electronic mail and files sent using FTP.

If you are submitting data to your technical support representative, use the following list (in order of most to least favorable) to determine the preferred method for submission:

- The preferred method of information submission is through FTP service over the Internet. If your environment supports FTP, you can place your file in the incoming directory on the host Cisco.com.
- The next best method is to send data by e-mail. Before using this method, be sure to contact your technical support representative, especially when transferring binary core dumps or other large files.
- Transfer through a PC-based communications protocol, such as Kermit, to upload files to Cisco.com. Again, be sure to contact your technical support representative before attempting any transfer.
- Transfer by disk or tape.
- The least favorable method is hard-copy transfer by fax or physical mail.

Note If you use e-mail, do not use encoding methods such as binhex or zip. Only MIME-compliant mail should be used.

INDEX

Numerics

A

B

C

D

E

F

G

H

I

J

K

L

M

Q

R

S

T

U

V

W